UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

TRABAJO FIN DE MÁSTER

Estudio del enrutado en la nube pública

Máster Universitario en Ingeniería de Telecomunicación

Autor: García Monsalve, Raquel

Tutor: García Dorado, José Luis
Ponente: López de Vergara Méndez. Jorge E.
Dpto. de Tecnología Electrónica de las Comunicaciones

JUNIO 2019

ESTUDIO DEL ENRUTADO EN LA NUBE PÚBLICA

AUTOR: García Monsalve, Raquel

DIRECTOR: García Dorado, José Luis
PONENTE: López de Vergara Méndez. Jorge E.

Dpto. de Tecnología Electrónica de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid

JUNIO 2019

Estudio del enrutado en la nube pública

Resumen

Actualmente, en la nube pública encontramos diversos servicios que han conseguido gran relevancia en el Internet. Sin embargo, es cierto, que no se conoce demasiado sobre el rendimiento que la nube ofrece a los usuarios. En este escenario, este Trabajo de Fin de Máster realiza un estudio sobre el enrutado de la nube pública, con el fin de analizar y conocer como enrutan los proveedores de servicios en la nube (CSPs). En concreto, conocer como son y como varían las rutas. Para ello, se ha contado con un conjunto de datos correspondientes a la toma de medidas de las trayectorias de paquetes, mediante la herramienta traceroute, entre múltiples centros de datos de varios CSPs durante cinco días.

Se ha realizado un análisis de los parámetros fundamentales de traceroute. Una vez conocido como funciona dicha herramienta, se ha analizado los datos y posteriormente se ha realizado un procesado de los datos para tener una mayor conocimiento a la hora de elaborar las rutas entre los diferentes centros de datos pertenecientes a los proveedores de servicios en la nube. En este procesado se analizan las direcciones IP que se encuentran en los datos, para así tener una base de datos de proveedores. Además, se ha realizado un sistema de búsqueda de la dirección IP, que devuelva a que operador pertenece, a través de la utilización del comando *Whois* desde la terminal Linux, previamente direccionado desde un script de Matlab.

Analizado los archivos que contenían las series temporales, se pasó a confeccionar las rutas, es decir, los sistemas autónomos que componen los saltos entre los centros de datos origen y destino. Para poder administrar de mejor manera las rutas entre todos los centros de datos en todos los instantes de tiempo, se ha realizado una tabla por cada instante de la serie temporal.

Finalmente, para observar la variación de las rutas en el tiempo se han utilizado diversas métricas, y se ha podido observar de esta manera si las rutas se mantienen estables en el tiempo o no. Además, se ha podido analizar cual es el porcentaje total de rutas que varían en la nube en general. Como también cuáles son los CSPs en los que se obtiene mayor variación en sus rutas, como dónde se encuentran ubicados los centros de datos que originan rutas más variables.

Palabras Clave

Cloud, rutas, proveedores de servicios en la nube, centro de datos, traceroute, Whois, distancia Levenshtein.

Study of routing in the public cloud

Abstract

Nowadays, we found different services in the cloud that have achieved great relevance. However, there is not much knowledge about performance that cloud offers to users. Therefore, this Master's Final Project gives a study about one of the keys of Internet performance, i.e., how routing on the public cloud is working. In particular, the aim is to analyze how cloud service providers (CSP) are routing and, especially, how such routes change over time. To this end, we had been provided with a rich set of Traceroute probes spanning measurements between different data centers in different CPS for five days.

After an analysis of fundamental parameters of Traceroute was carried out, the data have been analyzed and subsequently, processing of data has been performed to elaborate all the routes involving the set of data centers and CSPs. The processing of data analyzes IP addresses that are found in the data to have a provider database. This is made by searching which provider each IP belongs to, i.e., using "Whois" command from Linux terminal triggered from a Matlab script.

Once the files that have the temporary series are analyzed, the routes were constructed as a sequence of the Internet providers (ISP), i.e., the hops between origin and destination data centers. For managing routes between all data centers in all timestamps in an optimal way, a table for every timestamp of temporary series has been made.

Finally, by observing the variation of the routes over time, different metrics have been applied whereby it has been assessed whether the routes keep stable over time or not. Furthermore, we have also studied which is the total percentage of routes that changes in the cloud in general, which is the CSPs where get more variety in the routes and, likewise, where the data centers that originate more variable routes are located, among other comparisons.

Key words

Cloud, routes, cloud service provider, data center, traceroute, Whois, Levenshtein's distance.

Agradecimientos

En primer lugar, agradecer a mi tutor José Luis García Dorado, director del trabajo, por su ayuda y conocimientos que me ha proporcionado estos meses para la elaboración de este proyecto. Agradecer también a mi ponente Jorge E. López de Vergara.

Agradecer a cada una de las personas que me han apoyado durante la realización del proyecto.

Pero sobretodo agradecer a mi familia y amigos que han estado a mi lado en el desarrollo de este proyecto y durante toda la carrera apoyándome y animándome a seguir, sin ellos todo esto no hubiera sido posible.

Por último, agradecer a Carlos por estar en los buenos y en los malos momentos, y por su apoyo incondicional.

Índice general

Índice de figuras

Índice de tablas

1 Introducción

En este primer capítulo se tratará de presentar los aspectos que se van a desarrollar a lo largo de este estudio. En primer lugar, se muestra la motivación por la que surge el desarrollo de este proyecto.

A continuación, se presentan los objetivos que se pretende conseguir con este estudio y que fueron planteados antes del comienzo del proyecto. Por último, se muestran los puntos que se van a seguir en el desarrollo de esta memoria.

1.1. Motivación del proyecto

Actualmente, nos encontramos que los servicios que usan la nube pública (o comercial) están en continuo crecimiento [3]. Partiendo desde una persona que se encuentra en su casa y accede a Google docs, pasando por pequeñas empresas que usan microservicios como bases de datos, hasta empresas de distribución de contenidos que transmiten películas y otros recursos multimedia o grandes corporaciones que realizan *backups* y despliegan servicios web. Gran parte del uso se debe a lo extendida que está y su fácil acceso a través de Internet desde cualquier punto geográfico mientras exista una conexión a Internet.

Debido al gran éxito experimentado, la comunidad de Internet se centró en mejorar y monitorizar el rendimiento en la nube pública. De hecho, el reto que supone dar a los usuarios bajos tiempos de latencia ha sido tema de estudio en fechas muy recientes. En este sentido, frecuentemente, los centros de datos se encuentran distribuidos geográficamente como un intento de proveer una mejor calidad de servicio y una menor latencia para los usuarios finales [4].

Como resultado de esto, la comunidad de Internet ha demostrado interés en evaluar el rendimiento de las comunicaciones en la nube pública. Por ello, se han dedicado significativos esfuerzos en su monitorización. Obteniendo información relevante sobre su rendimiento y, así, calidad de servicio, importante tanto para los proveedores de servicio como para los clientes [5].

A pesar de las investigaciones de la nube centradas en la capacidad computacional [6], latencia, perdidas y ancho de banda [7], destaca que no se conoce demasiado sobre cómo se enruta en la nube pública. Incluso a pesar de que características como la longitud o el número de sistemas autónomos (AS) de las rutas usadas, así como los cambios en las mismas son clave para entender el rendimiento [8].

Durante todo este tiempo, como elemento fundamental de estas investigaciones ha sido sacar conclusiones lo más generales posibles, esto es, no caracterizar fenómenos concretos sino intentar capturar comportamientos generales en la heterogeneidad de Internet a lo que se ha llamado, modelos invariantes en Internet [9]. Este trabajo pretende seguir la misma línea, y buscar esos comportamientos generales o parcialmente homogéneos en la nuble pública.

En concreto, en este trabajo pretendemos dar luz a como se enruta y como varían las rutas en la nube pública. Para ello, vamos a estudiar un conjunto significativo de medidas de la herramienta traceroute centrada en los proveedores de servicios en la nube más populares tales como Amazon, Rackspace y Google. Traceroute es una herramienta que tanto investigadores como los proveedores de red han usado, históricamente, para poder diagnosticar algunos de los problemas de red y conocer más de la topología de Internet.

1.2. Objetivos

Una vez que se ha contextualizado el proyecto que se va a realizar y la motivación por lo que surge este estudio, se describirán los objetivos que se quieren conseguir.

El objetivo a alto nivel por el que se elabora este proyecto, es el de conocer mejor como enrutan los distintos proveedores de servicios en la nube (CSP) en el tiempo. Por tanto, el primer objetivo es caracterizar el enrutado de Internet, por ello, se necesita de determinadas medidas que nos permitan describir como viajan los paquetes entre los diferentes centros de datos en la nube. Para ello, se han analizado cientos de medidas de tipo traceroute entre 13 centros de datos pertenecientes a 3 CSPs durante 5 días. Estas medidas contienen los saltos (IP) que han dado los paquetes entre un centro de datos de origen en la nube y una dirección destino también perteneciente a la nube. Y una vez caracterizado el enrutado, tenemos como objetivo entender si las caracterizaciones de este fenómeno es válida para periodos largos de tiempo, o, al contrario, varían de forma significativa. Esto es, conocer mejor las variaciones del enrutado con el tiempo.

Para conseguir esto, es necesario primero plantear una serie de objetivos específicos. Estos objetivos son los que se muestran a continuación:

- Análisis exhaustivo de las series temporales de traceroute de los datos proporcionados velando que las medidas sean coherentes, válidas y estén presentes en todos, o al menos, en alguno de los CSPs. Además, un previo análisis de los parámetros fundamentales que devuelve la herramienta traceroute.
- Procesado de los ficheros que contienen las capturas de traceroute en particular desarrollando scripts en Matlab para el análisis de estos. También, se implementa un sistema que permita determinar a que sistema autónomo (AS) pertenece una dirección IP que se encuentra en la red.
- Construcción de rutas, entendiendo por ruta a la sucesión de sistemas autónomos entre un origen y un destino. Con esto, nuestro objetivo es construir a distintos

granos temporales las rutas entre todos los centros de datos estudiados para cada CSP. Con esta construcción de rutas, conoceremos los diferentes sistemas autónomos que se encuentran entre los centros de datos de origen y destino.

- Estudio de la variación de las rutas en el tiempo entre los diferentes centros de datos de cada CSP, teniendo en cuenta la latencia.
- Implementación de una métrica que compare y explique las diferentes tablas de rutas en el tiempo, y así poder detectar comportamientos generales y anómalos.

1.3. Planificación temporal

Para la realización de este trabajo, es necesario mostrar el tiempo que se ha invertido en cada una de las secciones. Éstas se componen de cuatro fases principales que a su vez se subdividen en tareas. La primera fase de ellas corresponde con la búsqueda de bibliografía y análisis de los conceptos fundamentales. En la segunda fase, nos centramos en el análisis de los datos que tenemos para el desarrollo del estudio y el diseño de los pasos que se van a seguir en la siguiente fase. En la tercera fase, se implementan las soluciones escogidas y se analizan los resultados. Por último, nos encontramos con la cuarta fase que se centra en la elaboración de la memoria y la defensa del trabajo. A continuación, se describen estas cuatro fases de manera más detallada:

Documentación sobre el proyecto (40 horas)

- Se establecen los objetivos que se quieren cumplir con la elaboración de este proyecto, y se realiza un análisis de éstos.
- Planificación temporal de las tareas a realizar en el estudio.
- Búsqueda de bibliografía relacionada con el trabajo que se va a realizar, además de la lectura de ella.
- Análisis de los conceptos fundamentales que se van a tratar en este trabajo sobre la nube. Desde las características de la nube abarcando hasta los tipos y modelos de ésta.
- Estudio de las posibles métricas que se pueden utilizar para la realización de la comparativa de rutas.

Diseño y análisis de los datos (65 horas)

- Análisis de los datos que se van a utilizar (capturas de traceroute), desde la herramienta que se ha utilizado para obtenerlos, como la cantidad de muestras que se tiene de cada centro de datos.
- Se concretan los pasos que se van a seguir para la realización del estudio, así como la puesta en marcha de herramientas necesarias para la implementación, en este caso la terminal de Linux en el sistema operativo Windows.

- Diseño de soluciones para los problemas encontrados en el desarrollo, uno de estos problemas es la falta de muestras en alguno de los instantes de tiempo.

Implementaciones y análisis de resultados (120 horas)

- Procesado de los datos y la elaboración de una base de datos de los operadores que nos encontramos en los datos.
- Implementación de un sistema que permite determinar el sistema autónomo al que pertenece una determinada dirección IP.
- Elaboración de las rutas entre los pares de centros de datos correspondientes a los CSPs.
- Elaboración de tablas de rutas como manera de organizar las rutas.
- Implementación de las métricas buscadas y comparativas estudiadas.
- Análisis factorial de los resultados obtenidos de las comparativas con el uso de funciones de distribución acumulativas.

Memoria y defensa (75 horas)

- Elaboración de la memoria de este proyecto en el que se incluye el fundamento teórico para el correcto entendimiento del trabajo, el análisis y procesado de los datos, las implementaciones realizadas, así como los resultados y conclusiones obtenidas con el desarrollo de este estudio.
- Elaboración y preparación de la defensa de este proyecto.

1.4. Organización de la memoria

En esta memoria queda reflejado el estudio e implementaciones que se han realizado a lo largo del desarrollo de este proyecto, para así, poder llegar a obtener conclusiones acerca del estudio del enrutado en la nube. A continuación, se muestran los capítulos en los que se ha dividido esta memoria.

- Capítulo 1. Introducción. En este primer capítulo se presenta el tema a tratar, la motivación por el que surge este estudio, se presentan los objetivos marcados que se deben llevar a cabo para la realización de este Trabajo Fin de Máster. Por último, se muestra la organización seguida para esta memoria.
- Capítulo 2. Estado del Arte. Se presenta el marco teórico relacionado con la nube, en este caso el Cloud Computing, sus características principales, ventajas y desventajas, tipos y modelos de "*cloud*". Para finalizar este capítulo, se presentan los estudios relacionados con el presente proyecto anteriores a la elaboración de éste.

- Capítulo 3. Datos y Metodología. En este capítulo se presentan los datos que se van a utilizar para la realización de este estudio. Como están formados dichos datos y el procesado que se realiza para determinar si los datos son coherentes y válidos para poder utilizarlos en este proyecto. A continuación, se mostrará el procesado que se ha realizado para analizar en profundidad las series temporales de traceroute que contienen los datos y así poder elaborar las rutas. Además, se mostrarán las métricas utilizadas para la comparación de los saltos IP contemplados en las rutas, y así, poder detectar el comportamiento que sigue el enrutado en la nube. Finalmente, se han mostrado los dos modelos de comparación de las métricas que se han utilizado en este proyecto.
- Capítulo 4. Resultados. Se mostrarán y analizarán los resultados que se han obtenido una vez aplicada las métricas a las rutas y se mostraran los comportamientos que se han observado en el enrutado. Además, se mostrarán cual es el mejor centro de datos origen y el peor centro de datos origen que enruta dependiendo la métrica utilizada. Finalmente, se destacaran las peores rutas que nos encontramos en los peores centros de datos origen que se mencionaron en este estudio.
- Capítulo 5. Conclusiones y trabajo futuro. En este último capítulo del estudio se presentan las conclusiones que se han extraído con la elaboración de este proyecto. Además, se expondrán las futuras líneas de trabajo a seguir en este ámbito y que creemos interesantes.
- Anexos. Al final del presente documento de este proyecto se añaden una serie de apéndices con información adicional para completar la información sobre este estudio.

2
Estado del arte

En este capítulo se desarrolla el estado del arte relacionado con el estudio que se presenta. En la primera sección se pretende introducir los fundamentos teóricos básicos que serán necesarios para la comprensión y motivación de este proyecto. En primer lugar, nos centraremos en la tecnología *Cloud Computing*, explicando sus características, ventajas y desventajas, tanto como sus modelos de servicios y sus tipos de Cloud que nos encontramos actualmente.

Se destaca la importancia y el crecimiento de este tipo de tráfico en nuestros días, y el que se prevé en los próximos años. A continuación, se recogen los trabajos previos relacionados con este proyecto. Revisando primero los estudios relacionados con la medición de la variación en la nube, desde el punto de vista de la latencia o utilizando diferentes métricas. Más adelante, se detallan los estudios que profundizan en la medición del ancho de banda en la nube.

2.1. Cloud Computing

Cloud Computing, aunque normalmente es denominado *cloud* o nube, es una importante tecnología que ofrece un conjunto de servicios, como pueden ser plataformas de desarrollo de software, servidores, almacenamiento y software a través de Internet con un modelo de pago determinado por su uso [10].

Las bases del Cloud Computing, a pesar de parecer un término relativamente actual, surgen en el año 1961 con John McCarthy [11] que expuso que "los avances en la información y las comunicaciones conducirían a que algún día la computación se organizaría como un servicio público", a modo de lo que conocemos hoy como una nube global. Además, predijo que la computación se convertiría en la base de una industria nueva e importante.

A partir de la década de los 90, se empieza a pensar que la red de Internet es capaz de soportar la implementación de la nube en su propia red, esto se debe a que Internet cuenta con un ancho de banda suficiente para poder soportarla. La compañía Salesforce [12], es la primera en ofrecer aplicaciones en la web, se le empezó a llamar a esto, computación en la

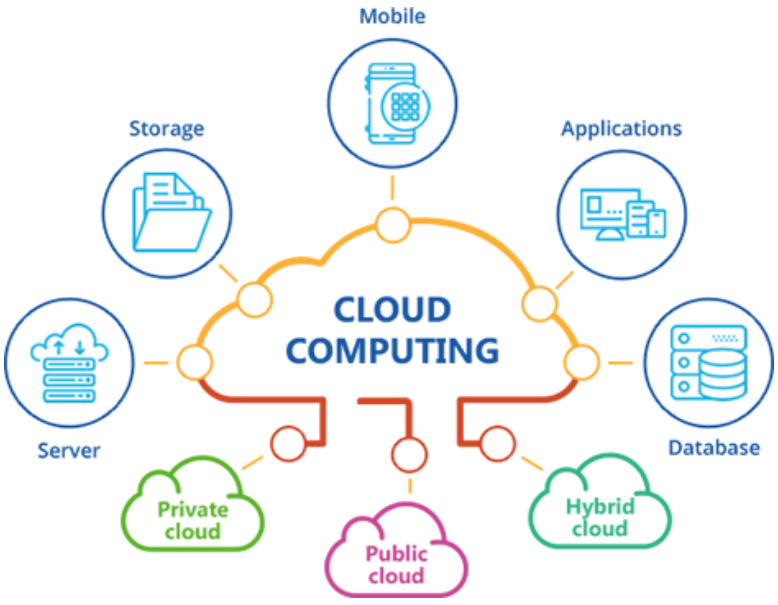


Figura 2.1: Cloud Computing [1]

nube. Años después en el 2002, Amazon desarrolla el sistema de almacenamiento, Amazon Web Services, germen de todos los serivicios en la nube que ofrece hoy en día Amazon. Posteriormente en 2006, lanzó EC2, Elastic Compute Cloud como servicio para pequeñas empresas y particulares. Debido al gran éxito que se obtuvo, en el 2009 tanto Google como Microsoft comienzan a proporcionar servicios en la nube.

2.1.1. Características

La tecnología del Cloud, destaca por ofrecer nuevas características en comparación con cualquier otro paradigma informático existente. A continuación, se exponen las principales características de esta tecnología [13]:

- Bajo demanda - Autoservicio. Un usuario de la nube dispone de capacidades de cómputo de manera independiente, como puede ser el uso de servidores y el almacenamiento en ellos, en función de las necesidades que tenga dicho usuario sin ser obligatoria la comunicación con el proveedor del servicio.
- Amplio acceso a la red. Los recursos que se encuentran ubicados en la nube, como puede ser el almacenamiento y las máquinas virtuales (VMs), se pueden acceder a ellas a través de la red de Internet. Por tanto, para poder acceder a los recursos se puede realizar desde cualquier ubicación geográfica que nos encontremos contando siempre con conexión a Internet, a través de de una interfaz de usuario a través de móviles, ordenadores, PDA.
- Asignación de recursos. Los recursos de computación, como puede ser procesamiento, memoria, ancho de banda, más los mencionados en las características anteriores, se agrupan de tal manera en un modelo de una sola instancia que es capaz de servir a múltiples usuarios. Esto lo consigue asignando dinámicamente los recursos físicos y virtuales, y que pueden volver a ser reasignados según la demanda de los usuarios finales.

- Rápida elasticidad y escalabilidad. Esta característica está relacionada con la anterior, dado que las capacidades que se necesitan se asignan o se liberan de manera elástica, casi en tiempo real. Y como bien se ha dicho de manera automática, casi sin interacción con el proveedor del servicio, para que así se escale y amplíe rápidamente. Un ejemplo de esto, lo encontramos en una aplicación que estemos utilizando al 20 % pero al día siguiente necesitamos su uso al 90 %, esto se notifica al proveedor y se cambia la tarifa de costo automáticamente.

- Servicio medido [14]. Los recursos que utilizan los usuarios en la nube son monitorizados y controlados por los CSPs con el modelo de pago ya mencionado anteriormente. Esto es comparable a los servicios que utilizamos a diario como puede ser la electricidad o el agua.

2.1.2. Ventajas y Desventajas

Algunos de los beneficios que proporciona la tecnología de la nube, se han podido intuir con la presentación de las características que cuenta la tecnología y que se han mencionado en el apartado anterior, pero a continuación se destacan dichas ventajas [12]:

- Ahorro en los costes: Los usuarios finales en la nube, únicamente tienen que pagar por lo que consumen de los servicios que utilizan. Además, la inversión inicial es mínima debido a que no requiere que compren infraestructura. El costo de mantenimiento también es bajo, esto se debe a que los servicios no se encuentran en el ordenador de cada usuario, sino que para acceder a ella necesitan de la red de Internet dado que se encuentra ubicado en la nube, por tanto, los servicios no tienen que ser actualizados en el ordenador de cada uno de los usuarios.

- Flexibilidad: Como bien se ha mencionado en las características de la nube, se ha destacado lo escalable que es ésta, para poder adaptarse rápidamente a los cambios que se necesitan en ella para adquirir los recursos, tanto si se necesitan más como si se necesitan menos recursos. Por tanto, para poder hacer frente a estas variaciones la nube tiene la ventaja de ser flexible.

- Seguridad: El uso de la nube garantiza una mayor seguridad, dado que esta utiliza datos cifrados, el acceso se realiza a través de ciertos controles sólidos y en los que es necesario el uso de claves. Destacar, que el proveedor es el que debe garantizar la seguridad de la infraestructura.

El paradigma de computación en la nube aún sufre de algunas desventajas que intentan ser estudiadas por investigadores para intentar en un futuro subsanarlas. A continuación, se muestran algunas de ellas [15]:

- Seguridad. Algunos de los usuarios de la nube dudan de la seguridad de ésta, debido a que no conocen si la información está más segura en sus propios recursos físicos o en la nube. Además, en la nube los datos se distribuyen sobre la red, sin importar dónde se guarden realmente los datos perteneciente a los usuarios. Existen estadísticas que muestran que un tercio de las infracciones se deben a dispositivos personales robados o perdidos, y un 16 % se debe al robo de información sensible que ha sido expuesta por los empleados.

- Fiabilidad. La nube destaca por la fiabilidad que ofrece que es constante. Pero últimamente han surgido algunos casos en los que la nube no estuvo disponible durante unas horas. Con ello, se puede observar que la nube sufre algunos problemas al igual que lo puede sufrir un propietario, es decir, que muestra tiempos de inactividad. Se quiere que en un futuro, los estándares estén establecidos y se tengan servicios más ricos. Finalmente, los servidores en la nube experimentan tiempos de inactividad, caída del servidor, y velocidades bajas, pero la diferencia reside en que los usuarios mantienen una mayor dependencia del proveedor de servicios en la nube (CSP).
- Privacidad. Destacar que la nube al utilizar la computación virtual, puede ocurrir que los datos personales de los usuarios se encuentren dispersos entre diferentes centros de datos, en lugar de permanecer siempre en una misma ubicación física, debido a la dispersión puede suceder que traspase las fronteras nacionales. En el momento, en el que la información traspasa las fronteras, la privacidad de dichos datos ya no se enfrenta a sistema jurídico nacional, sino que se enfrenta al de otros países. Otras maneras de incumplir la privacidad, es que los usuarios filtren la información oculta cuando acceden a los servicios en que se encuentran en la nube.
- Rendimiento. Una de las desventajas de la nube puede ser que no consiga un rendimiento adecuado para cierto tipos de aplicaciones que se ejecutan en ella, como puede ser aplicaciones de transacciones y otras aplicaciones de datos. Otro escenario que influye en el bajo rendimiento de la nube es que los usuarios se encuentran a una gran distancia de los proveedores de la nube, provocando una alta latencia.

2.1.3. Modelos de Servicios

En la nube nos podemos encontrar con diferentes servicios de computación. Estos se clasifican, a alto nivel, en tres niveles [16], infraestructura como servicio (IaaS), plataforma como servicio (PaaS) y software como servicio (SaaS). A continuación, se muestra más de información sobre cada uno de ellos.

- Infraestructura como Servicio (IaaS) [13]. Es un paradigma por el cual se proporciona recursos en *bruto*: esto incluye el procesamiento, el almacenamiento y otros recursos hardware de forma virtual. Estos recursos son administrados a través de la red de Internet. Los recursos no se compran o alquilan de la manera tradicional, lo que sería pagando cuando se compran o mediante tarifas mensuales, sino que se paga por lo que se usa según las necesidades del usuario de la nube. Un ejemplo de la infraestructura que se puede alquilar son las máquinas virtuales, las cuales se puede escalar rápidamente en función de su uso, y todo ello gracias a los avances de la virtualización. Los usuarios no tienen control real sobre la infraestructura de la nube, pero si tienen el control sobre los sistemas operativos, el almacenamiento y aplicaciones que se da sobre ella. Los ejemplos que encontramos sobre este tipo de infraestructura son Elastic Cloud Computing (EC2) que permite manejar máquinas virtuales en la nube y Simple Storage Service (S3) que actúa como servicio de almacenamiento, en ambos casos son de acceso público.
- Plataforma como Servicio (PaaS) [13]. Proporciona al usuario un entorno de programación y ejecución. Por tanto, el usuario de esta plataforma puede crear aplicaciones con diferentes lenguajes de programación que sean compatibles con el proveedor hasta se puede implementar en la infraestructura del proveedor. En este tipo de servicio

el usuario no puede controlar la infraestructura de la nube como en el caso anterior, pero si se tiene el control de las aplicaciones implementadas en ella. Una de las ventajas que se producen es que se reduce la carga de administración del sistema (cambios entre los diferentes entornos: desarrollo, pruebas y producción), por tanto, gracias a esto se pueden centrar en otros problemas que surgen en el desarrollo. Como ejemplo, nos encontramos con Google App Engine, con el que se pueden crear aplicaciones en los sistemas. Además, hace que el desarrollo de aplicaciones sea más fácil, dado que se puede obtener ayuda de otros desarrolladores del mundo.

- Software como Servicio (SaaS) [13]: Este tipo de servicio proporciona a los usuarios de la nube aplicaciones completas a través de la red de Internet. Por tanto, estas aplicaciones se encuentran en la nube y los usuarios pueden acceder a ellos a través de navegadores. Con ello, surge la gran ventaja de que no es necesario instalar, ni mantener dicha aplicación en los propios ordenadores. En este caso son necesarias las políticas de autenticación para así poder separar los datos de los usuarios. Además, nos encontramos con el usos de aplicaciones complejas como son los ERP (planeamiento de recursos empresariales) y CRM (administración de las relaciones con el cliente). Como ejemplo donde encontramos este tipo de servicio es con Google Docs y Office 365 e incluso el correo electrónico, dado que para acceder a él es necesario que el usuario inicie sesión a través de Internet.

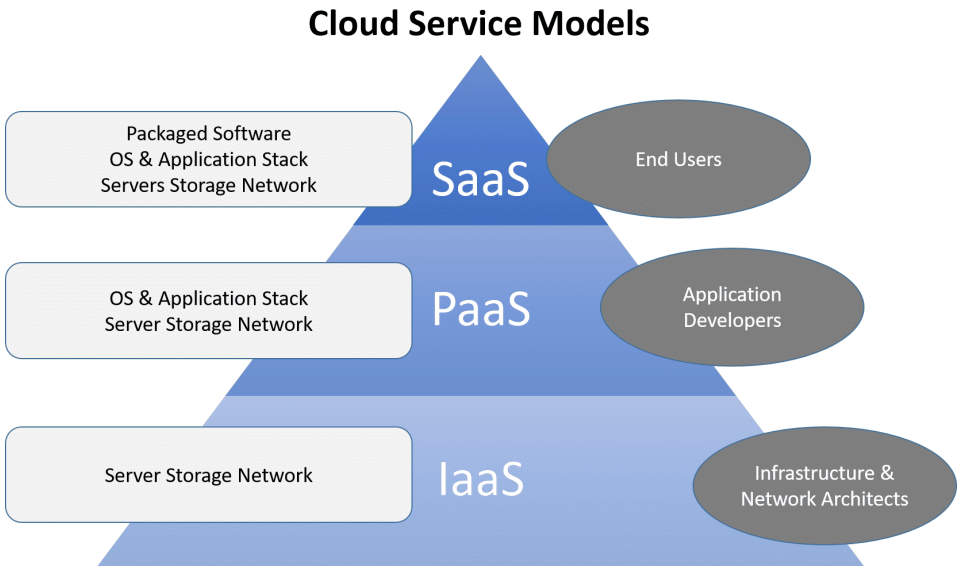


Figura 2.2: Modelos de Servicios [2]

2.1.4. Tipos de Cloud

En la nube nos encontramos con diferentes tipos de clouds, estos son posibles clasificarlos según la relación que existe entre el proveedor y los usuarios. A continuación, se presentan los tipos de Cloud [17]:

- Cloud público. Este tipo de nube cuenta con la infraestructura y los recursos que se ubican en la nube, el público en general es quien cuenta con acceso a ellos. Por

tanto, la nube está al servicio de un conjunto amplio de personas a través de la red de Internet. Es el tipo de Cloud en el que se centra este trabajo.

- Cloud privado. En este caso la nube es utilizada en exclusividad por una empresa, o algún tipo similar, en la que tiene acceso único múltiples personas que pertenecen a ella. Este puede ser administradas por la propia empresa o por empresas de terceros.
- Community Cloud. Este tipo de nube, se ofrece a un conjunto de personas específicas que comparten inquietudes similares. Este tipo como en el caso de la nube privada, puede ser gestionada por ellos mismos o por terceros.
- Cloud híbrido. Como su nombre indica es la combinación de varios tipos de nubes, desde dos o más nubes de las mencionadas anteriormente. Cada nube permanece como una entidad única pero que está unidas por cierta tecnología. Un ejemplo de ello lo encontramos en el equilibrio de cargas entre nubes.

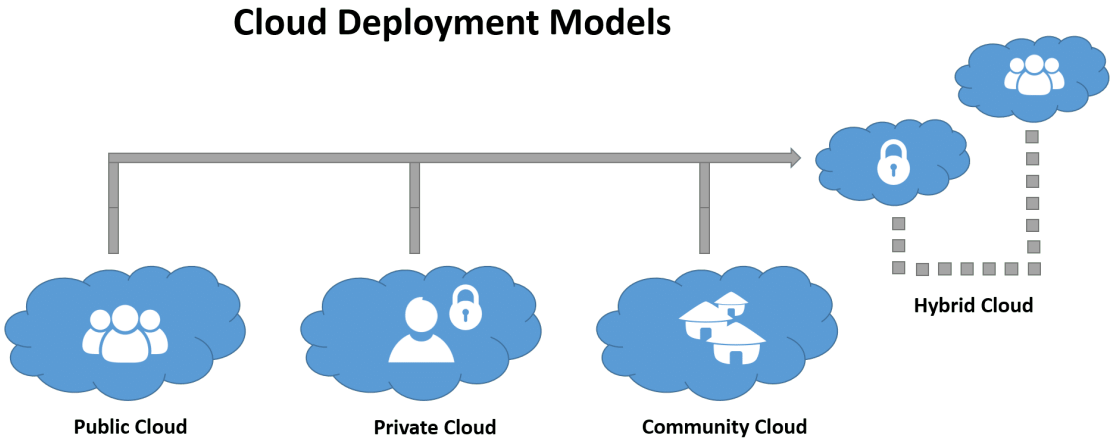


Figura 2.3: Tipos de Cloud [2]

2.2. Trabajos Relacionados

Una vez que se han mostrado las bases teóricas de la tecnología de computación en la nube, se recogen los trabajos relacionados con este estudio. En nuestro caso nos centraremos en trabajos relacionados con las mediciones que se han realizado en la nube con diferentes métricas y así poder observar el rendimiento de la nube. Además, nos centraremos en el ancho de banda, dado que el ancho de banda que se mide entre los CSPs, puede estar limitado por las rutas que se siguen entre entre ellos.

En 2005, se midió el ancho de banda entre la mayoría de los nodos de la infraestructura Planetlab [18]. Planetlab es una plataforma distribuida globalmente con cientos de nodos a lo largo de más de 25 países. Con él, se pueden realizar diversos experimentos de Internet y probar cualquier idea o algoritmo. En este caso, los autores aprovecharon la distribución de los nodos de Planetlab para medir el ancho de banda entre ellos. Aunque hay más de 500 nodos implementados en Planetlab, solo un poco más de la mitad respondieron a la campaña de medida. Los autores decidieron utilizar Pathrate, una herramienta que estima la capacidad máxima de las rutas en Internet, para realizar las mediciones de

ancho de banda. Pero a la hora de analizar los resultados que obtuvieron se encontraron con algunos límites. Las capacidades para algunos nodos estaban limitadas y no ofrecen información significativa de la red, sino de la capacidad limitante. No obstante, a pesar de estos problemas, obtuvieron medidas de ancho de banda entre 80 Mb/s y 120 Mb/s.

Dentro del escenario de estudio del caso anterior, destaca una gran motivación por comprobar cómo funciona Internet. Y dentro del Internet, destaca la importancia y relevancia que tiene hoy en día la computación en la nube. Destacar que el tráfico Cloud ha experimentado un crecimiento importante a lo largo de todos estos años. Aunque no solo ha crecido estos años anteriores, sino que se prevé que aumente el triple el tráfico Cloud para el 2021 [3]. Dichos datos se puede observar en la figura 2.4. En ella se puede ver que la cantidad de tráfico anual en 2016 fue de 6.8 ZB y para 2021 se triplicará hasta alcanzar los 20.6 ZB por año.

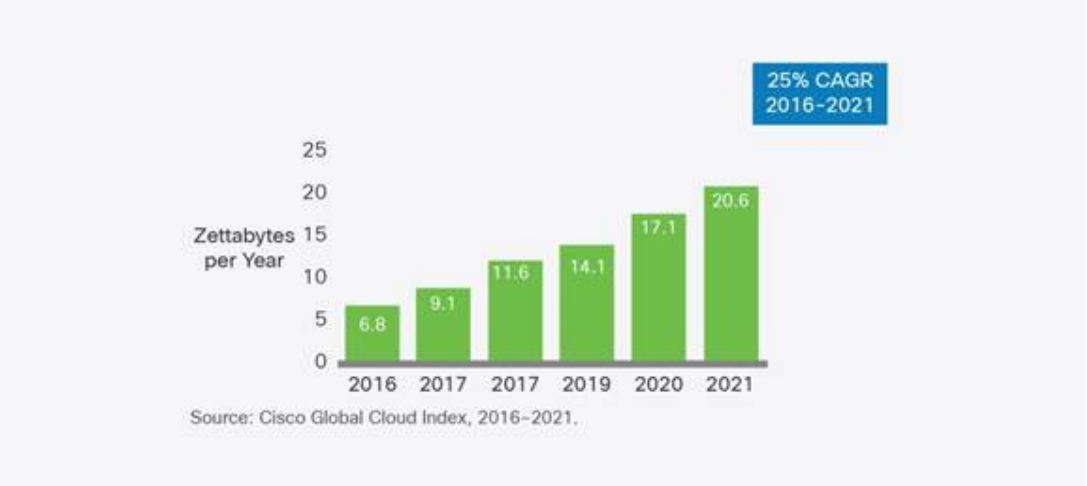


Figura 2.4: Crecimiento del tráfico Cloud [3]

Debido al gran crecimiento que ha experimentado el tráfico Cloud, despertó interés en los investigadores, centrándose en estudiar y analizar dicho tráfico. A continuación, destacamos algunos de los trabajos de estos investigadores que han estudiado el rendimiento y variación del tráfico Cloud.

Se estudió el rendimiento de Amazon [19], uno de los pioneros en ofrecer servicios en la nube, con el que se pretende guiar a los proveedores para futuras actualizaciones y mejoras. Para este estudio se realizaron medidas de latencia y de rendimiento de la CPU, con el que se llegó a la conclusión de que los proveedores en la nube deberían aprovisionar redes de alto rendimiento, dado que el rendimiento no era satisfactorio en programas paralelos con uso de memoria distribuida o compartida. Amazon, no solo ha sido estudiado desde la perspectiva de los servicios y la infraestructura, también ha sido examinada desde el punto de vista de los usuarios finales. Este estudio se realizó con usuarios en Italia y tomaron medidas pasivas [20]. Con estas medidas concluyeron que, el centro de datos de Virginia es el responsable de un 85 % del tráfico enviado a los usuarios finales en Italia, en lugar de realizarse con el centro de datos de Irlanda que tienen disponible, que es más cercano a este país. Además, concluyeron que el rendimiento que muestra el centro de datos de Virginia era bajo.

En 2010, se presentó una caracterización de la nube de los diferentes proveedores. Para ello, los autores desarrollaron una herramienta llamada CloudCmp [21] y la usaron

para evaluar a algunos proveedores de la nube que dominan el mercado. Se centraron en métricas importantes como, la velocidad de CPU, memoria, latencia de red, tiempo de respuesta y ancho de banda disponible. Con ello, consiguieron herramientas individuales que son simples, pero en conjunto permiten comparar los proveedores en la nube de manera novedosa. Estudiaron la red intra-centro-de-datos (conexión dentro del mismo centro de datos), que a veces tiene propiedades diferentes respecto con la red inter-centro-de-datos (conexión con otros centros de datos). Concluyeron que tienen menor latencia la red intra en términos de RTT con respecto a la red inter.

En la misma línea, encontramos un análisis de la variación del rendimiento de medidas como la capacidad CPU, memoria y la velocidad de escritura y lectura, en proveedores públicos de IaaS [6]. Como conclusión final obtuvieron que la región en la que se encuentra el centro de datos tiene un impacto significativo en el rendimiento.

Dentro de los estudios que medían el rendimiento en términos de conectividad dentro de la nube, la mayoría de los trabajos se han centrado en medir el latencias mediante el RTT debido a que es de bajo precio dado que la nube pública factura las transferencia de datos por volumen. Sin embargo, más recientemente está empezando a coger gran relevancia entre otros términos, el ancho de banda, que es fundamental en términos de rendimiento. Destacando la importancia de este término en la transferencia de datos de los dispositivos que se encuentran conectados parta realizar backups, clonar máquinas virtuales, distribuir sistemas operativos o replicar contenidos multimedia [22].

En 2015 se publicaron dos estudios en los que se mostraban mediciones activas de la nube en una única región (intra-cloud). En el primer estudio, los autores se centraron en Microsoft Azure [23], uno de los principales proveedores en la nube pública. Y su siguiente estudio se centró en otro de los principales proveedores, Amazon Elastic Compute Cloud (EC2) [7]. Y midieron el ancho de banda en diferentes escenarios. En el primer estudio tomaron medidas durante 800 horas mientras que en el segundo estudio se incrementó a 5000 horas de experimentos. Para la medición del ancho de banda, se midieron en diferentes escenarios con diferentes tamaños de máquina virtual, diferentes regiones, de protocolo de transporte, mecanismos de direccionamiento. Los resultados de anchos de banda que se obtuvieron en estos diferentes escenarios van de cientos a miles de Mb/s. Además, verificaron que las máquinas virtuales no pueden inyectar tráfico a la red a una velocidad superior al umbral contratado, y este dependerá directamente del tamaño de la máquina virtual. Por lo tanto, quedó demostrado con este estudio que el ancho de banda es dependiente del tamaño de la máquina virtual, mientras que la región geográfica donde se han realizado las medidas tiene una influencia bastante limitada.

También en 2015, se realizó un estudio sobre Amazon EC2 de múltiples aplicaciones [24], es decir, se combinaban diferentes CPUs y máquinas virtuales para que coincidieran con los requisitos de diferentes aplicaciones en la nube y así poder observar el rendimiento que podría darse en la nube. Consideraban que el ancho de banda es uno de los requisitos más importantes que deberían cumplir estas aplicaciones en la nube, y por tanto, se midió el ancho de banda entre una serie de diferentes CPU y máquinas virtuales en Amazon EC2, tanto intra-centro-de-datos como inter-centro-de-datos, durante un tiempo de 19 meses. Los resultados que se mostraron es que hay una gran variabilidad en el ancho de banda tanto para intra-centro-de-datos, como para inter-centro-de-datos. En intra-centro-de-datos las VMs de menor tamaño encuentran su ancho de banda limitado en 300 Mb/s, pero en los requisitos que mayor ancho de banda se consigue son en VMs más grandes que consiguen llegar a 1 Gb/s cuando ambos extremos se encuentran en el mismo centro de datos . Para el caso inter-centro-de-datos, se obtuvieron resultados de anchos de banda demasiado bajos,

por debajo de 200 Mb/s.

En el 2016, comienza a centrarse el interés en el estudio del ancho de banda de manera inter-cloud, en este caso se evalúan entre los proveedores de Amazon y Azure los términos de ancho de banda y latencia [25]. Realizaron el estudio sobre unas 300 horas de tráfico en 12 combinaciones para cada región. Quedando demostrado que la infraestructura entre centros de datos Azure funciona mejor que Amazon en términos de ancho de banda. Además, se observó que el ancho de banda es variable dependiendo de la región en la que se encuentre. Respecto al ancho de banda para TCP, se obtienen resultados mucho más bajos, no superando los 300 Mb/s. Mientras que para UDP se obtienen valores más altos entre 600 y 800 Mb/s.

En el año 2017, alguno de los autores anteriores decidió profundizar en el estudio realizado en los años anteriores, y esta vez realizaron mediciones alrededor de unas 800 horas [26]. Se destaca la gran infraestructura desplegada de alto rendimiento entre los centros de datos de la nube para los proveedores de Amazon y Azure, ambos son capaces de llegar a un ancho de banda de 800 Mb/s (UDP) entre centros de datos distribuidos geográficamente. Aunque con TCP se sigue sin conseguir grandes resultados de ancho de banda.

En otro estudio donde se comparan series temporales de ancho de banda de medidas inter-cloud es [4], modelándolas de forma factorial. Aquí obtienen la conclusión de que las cifras de ancho de banda que se pueden conseguir varían entre centros de datos y CSPs según la localización. Además, encontraron que las bajadas de rendimiento en ancho de banda no estaban correladas en la nube pública reforzando la idea de robustez en la nube. Esto es, en resumen, que un centro de datos no vaya bien, no significa que toda la nube vaya mal, y por tanto podemos usar otros centros de datos como sustituto.

Finalmente, en nuestra búsqueda de información relacionada con nuestro proyecto sobre el estudio del enrutado en la nube pública, nos hemos dado cuenta que no ha sido previamente abordado a pesar de la gran relación que mantiene las rutas entre los diferentes proveedores de servicio con el rendimiento.

3 Metodología

En este capítulo se presentan los datos a los que hemos tenido acceso para la elaboración de este proyecto. En primer lugar se realizará un análisis exhaustivo de ellos. Este análisis se realiza con el fin de poder garantizar que los datos que vamos a utilizar para este estudio sean coherentes y válidos. Posteriormente, se irá explicando en detalle el procesado que se ha realizado para obtener finalmente las rutas entre los diferentes centros de datos de cada CSP.

Finalmente, se explican las métricas utilizadas y las opciones que hemos creído conveniente para poder realizar el estudio del enrutado en la nube pública.

3.1. Datos

Los datos disponibles para la realización de este estudio corresponden a un conjunto de medidas realizadas mediante máquinas virtuales que son lanzadas en los correspondientes CSP y centros de datos en estudio. Con estas medidas, posteriormente, fue posible realizar diferentes estudios, en concreto sobre el enrutado en la nube pública.

En los estudios que se han mencionado en el capítulo anterior, la mayoría se centraban en estudiar los cuatro CSPs más populares, que son: Microsoft Azure, Rackspace Cloud, Google Cloud y Amazon. Destacando que hoy en día estos cuatro, se han convertido en los proveedores dominantes en la nube.

Destacar que los datos que disponemos corresponden a los cuatro proveedores de servicios mencionados. En cada uno de los CSP, tenemos una serie de centros de datos (DC), resultando un total de 18 puntos de presencia (PoP) distribuidos geográficamente. La distribución geográfica de los CSPs se puede observar en la figura 3.1. En cada uno de estos PoP se inició una máquina virtual de alta capacidad. Además, en la tabla 3.1, se puede observar el nombre de cada uno de los centros de datos, el área geográfica y el proveedor al que pertenece.

Para el proceso de medición entre los diferentes centros de datos que se encontraban distribuidos geográficamente, se usa CloudB [4]. Esta plataforma pensada en concreto para

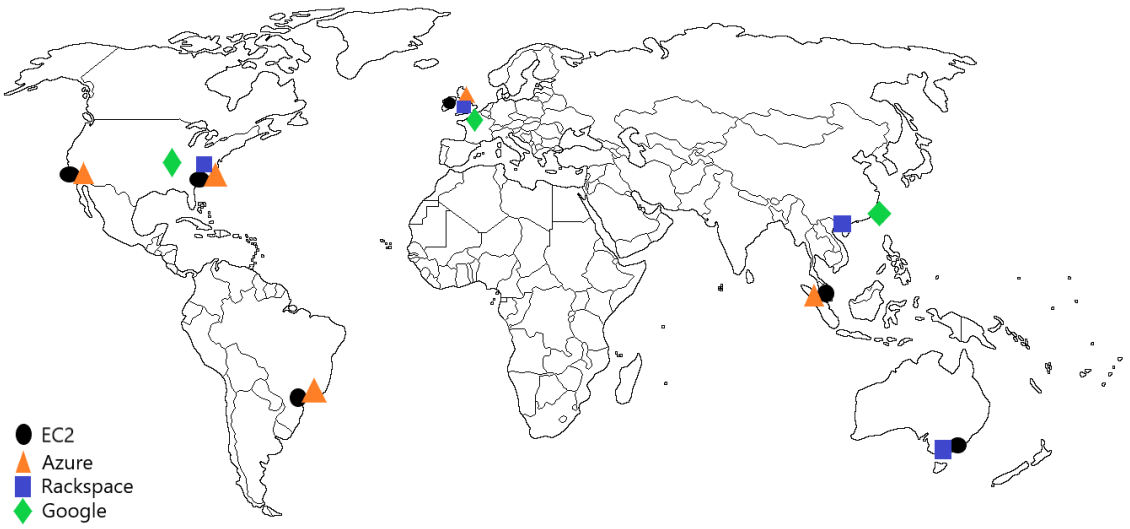


Figura 3.1: Distribución geográfica de los centros de datos

CSP	Área Geográfica	Nombre Centro de Datos
Amazon EC2	Eastern US	Virginia
	Western US	California
	Nothern Europe	Ireland
	East Asia	Singapore
	Australia	Sydney
	South America	Sao Paulo
Microsoft Azure	Eastern US	Virginia
	Western US	California
	South America	Brazil
	Nothern Europe	Dublin
	East Asia	Singapore
	Eastern US	Virginia
Rackspace Cloud	Nothern Europe	London
	East Asia	Hong Kong
	Australia	Sydney
Google Cloud	Central US	Iowa
	Nothern Europe	Belgium
	East Asia	Taiwan

Tabla 3.1: Centros de Datos (CSPs)

medidas en la nube consiste en un planificador de ejecución de herramientas de medidas en las que su duración y frecuencia son configurables. CloudB, con esta configuración, ejecuta todas las herramientas entre las direcciones IP (origen y destino) incluidas en la lista de IPs a estudiar, en este caso los centros de datos. CloudB funciona creando scripts auxiliares para cada una de las herramientas que se quiere utilizar, incluyendo los parámetros correspondientes de entrada. Finalmente, autoconfigura el planificador de tareas cron de UNIX con la frecuencia que se requiere para la ejecución de las herramientas y exporta los resultados en texto plano. En nuestro caso, la herramienta que nos interesa es traceroute que está incluida por defecto en CloudB.

Finalmente, se obtuvo el banco de pruebas que contenía los resultados de la herramienta traceroute, correspondiente a los 18 centros de datos que se ha podido observar en la tabla 3.1, que agrupándolos en rutas de origen y destino se obtiene un total de 306 rutas. Los datos que se van a utilizar, están formados por medidas de traceroute entre origen y destino de los centros de datos pertenecientes a los CSP cada hora durante 5 días. Por tanto, en cada archivo se podrá observar las trazas de traceroute durante 5 días entre un origen y un destino. Un ejemplo de este archivo se puede observar en el Anexo A.

Antes de adentrarnos por completo en el análisis de los datos, es fundamental conocer acerca de la herramienta de traceroute, cómo funciona y qué parámetros devuelve. Traceroute intenta averiguar la ruta que sigue un paquete IP hasta el host destino que se encuentra en la red de Internet. Traceroute [27] utiliza el campo Time To Live (TTL) del protocolo IP, este campo sirve para que un paquete no esté en la red de forma indefinida. El TTL es un número entero que va decreciendo en función de los nodos por los que pasa el paquete. Cuando el campo TTL llega al valor 0 ya no se reenvía más el paquete, lo descarta y envía una respuesta ICMP de tiempo excedido al origen donde se lanzó traceroute.

Por tanto, traceroute empieza mandando paquetes a la red, de forma que el primer paquete enviado lleva el valor del campo TTL a 1, el segundo paquete enviado el campo TTL es 2. Así, el primer paquete será eliminado en el primer nodo, y mandará al origen un mensaje ICMP. El segundo paquete enviado pasará el primer nodo y llegará al segundo nodo, donde será descartado, devolviendo un mensaje ICMP al origen. Esto se realiza sucesivamente hasta llegar a la dirección IP destino.

Destacar que para cada valor de TTL se envía tres paquetes (o sonda de prueba). El mensaje ICMP que devuelve en cada nodo muestra el valor de TTL, la dirección de puerta de enlace y el tiempo de ida y vuelta de cada sonda (RTT).

Por tanto, en cada muestra de traceroute se puede observar que la primera fila devuelve el día y la hora a la que es lanzado traceroute. En las líneas sucesivas, se muestra la dirección IP de cada nodo por los que va pasando los paquetes. Como bien se ha indicado se envían tres sondas, por tanto, en cada línea pueden aparecer hasta tres direcciones IPs diferentes si siguen caminos distintos los paquetes. Si siguen el mismo camino solo se muestra una, o puede surgir que solo un paquete vaya por diferente camino. Además, se observará el tiempo RTT de cada uno de los paquetes a dicho nodo correspondiente. Para la respuesta del siguiente nodo se observará en la siguiente línea. Por último, si un nodo no responde se devolverá un asterisco (*).

A pesar de que las medidas fueron realizadas para los cuatro proveedores dominantes, cabe destacar que se ha tenido que excluir al proveedor Microsoft Azure de los datos, dado que traceroute, como bien ya se ha comentado, utiliza el protocolo ICMP y los centros de datos de este proveedor bloquean dicho tráfico por motivos de seguridad [28]. Por tanto, fue eliminado de nuestro banco de pruebas, quedando finalmente 13 centros de datos pertenecientes a 3 CSPs y un total de 156 rutas. A continuación, se observa en la tabla 3.2, los centros de datos de los CSPs de los datos finales con sus correspondientes direcciones IPs.

CSP	Dirección IP	Nombre Centro de Datos
Amazon EC2	54.85.105.226	Virginia
	54.183.83.172	California
	54.76.187.119	Ireland
	54.251.141.201	Singapore
	54.79.4.33	Sydney
	54.207.116.46	Sao Paulo
Rackspace Cloud	104.130.3.144	Virginia
	134.213.50.86	London
	119.9.88.108	Hong Kong
	119.9.23.77	Sydney
Google Cloud	146.148.57.172	Iowa
	130.211.77.173	Belgium
	107.167.190.83	Taiwan

Tabla 3.2: Centros de datos utilizados

3.2. Análisis de datos

Una vez introducido el banco de datos que tenemos para la realización del estudio y conocer dónde están ubicados los centros de datos con los que se va a trabajar, es necesario realizar un análisis exhaustivo de las series temporales que contienen los archivos de traceroute, para así poder garantizar que las medidas son coherentes, válidas y están presentes en todos los CSPs.

Realizando el análisis mencionado nos enfrentamos al primer problema. Los archivos que contienen las series temporales, puede ser que les falte alguna muestra o que cada archivo tenga un número diferente de series temporales. Esto quiere decir que en las muestras pertenecientes a un determinado día pueden faltar algunas horas en las medidas que se realizaron, por ejemplo, que pasen las medidas de las 10:00 a las 16:00. Por ello, antes de comenzar a realizar el procesado de los datos, se ha decidido contabilizar todas las muestras que contiene cada archivo de las series temporales.

Anteriormente, se ha explicado cómo funciona traceroute dado que es necesario para poder analizar los datos que se nos han proporcionado. El procesado que se ha realizado para este análisis es: se empieza contabilizando el número de muestras entre cada origen y destino de los centros de datos, teniendo en cuenta para ello la primera línea que devuelve traceroute que contiene la fecha y hora de la muestra. Esta información facilita el proceso de análisis. Una vez contabilizado el número de muestras entre cada origen y destino, se busca el que tiene el mayor número de muestras entre todos los archivos y con ese número calculamos el porcentaje de muestras correspondiente a cada archivo de muestras.

En la figura 3.2 se visualizan los porcentajes de datos disponibles para cada ruta, en ellos se puede observar que la mayoría de los archivos tienen un número razonable de muestras, llegando casi al 100 %. Pero el problema surge con los archivos pertenecientes a los centros de datos de Google contra los centros de datos de los otros proveedores, que tienen un bajo número de muestras, se pueden observar esos cuadros de un tono azul más claro, pero hemos decidido seguir utilizando esas muestras para posteriormente ver qué ocurre con los resultados que obtenemos de dicho caso, aunque es un factor a tener siempre presente en la realización del análisis de los resultados.

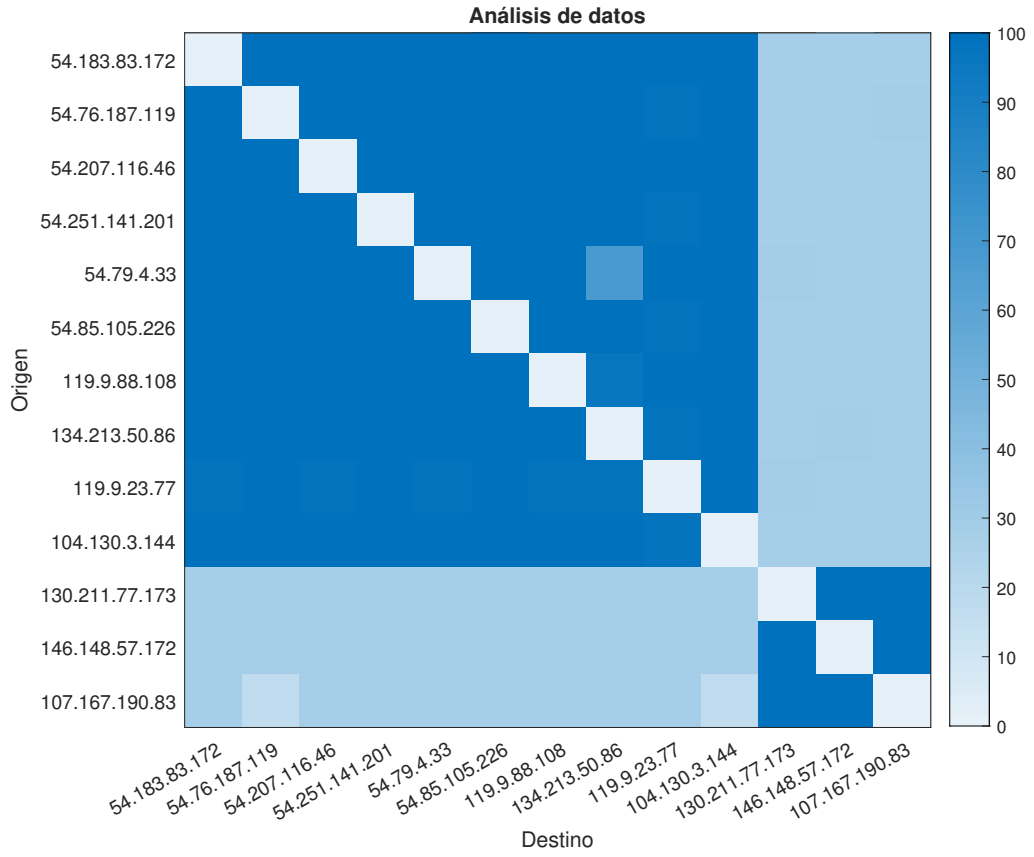


Figura 3.2: Porcentaje de muestras en los datos

3.3. Limitación de los datos

Para el problema que nos hemos encontrado de la falta de algunas muestras en los archivos de las series temporales se ha decidido duplicar muestras para que no haya huecos. En este caso, la muestra que se duplicará es la anterior, es decir, si tenemos la muestra de la hora 13:00 y salta a la muestra de la hora 15:00, se duplica la muestra de las 13:00 haciéndola pasar por la muestra de las 14:00.

Para ello, como bien se ha comentado anteriormente, en la primera línea que devuelve la herramienta de traceroute, viene indicada la fecha y hora, por tanto, se guarda la de la primera muestra y se compara con la fecha y hora de la siguiente muestra, si hay más de un salto entre ellas se duplica la información de la anterior muestra. Con esto conseguimos obtener unos datos coherentes y válidos para poder continuar con el estudio del enrutado en la nube.

Otro factor que se ha tenido en cuenta es, que si una muestra faltaba en todos los archivos del banco de pruebas que ha sido proporcionado, se ha decido no duplicar esa muestra. Ya que no tiene sentido crear dicha muestra si hay que añadirla en todos los archivos, estamos añadiendo muestras que no aportan información.

3.4. Análisis de Proveedores de Servicios

Para poder continuar con el desarrollo del estudio, es necesario de antemano conocer la cantidad de proveedores y de AS que se encuentran entre nuestros datos, para así poder facilitar la construcción de las rutas.

Antes de continuar con el análisis, cabe resaltar que entendemos por ruta. Consideramos que una ruta es una sucesión de proveedores por los que pasa un paquete en la red, entre un origen y un destino. Pero otro detalle a tener en cuenta para la realización de este estudio es, como bien ya se ha mencionado, traceroute devuelve la dirección IP de la interfaz de los routers por los que pasa dicho paquete.

A cada proveedor le pertenece un rango de direcciones IP, con ello, surge el problema de que hay que ubicar la dirección IP de cada router o nodo que ha sido devuelto por traceroute a qué proveedor pertenece. Por tanto, surge la necesidad de realizar una traducción de dicha dirección IP obtenida a que proveedor corresponde.

Para realizar esta traducción, ha sido necesario buscar en todos los archivos que se tienen disponibles y procesar cada dirección IP a que proveedor corresponde. Para ello, ha sido necesario ir recorriendo uno a uno los archivos y en cada una de las muestras se ha cogido una a una cada dirección IP, antes de nada teniendo en cuenta que un salto en traceroute puede tener diferentes direcciones IP, hasta un total de tres, ya que cada paquete puede tomar caminos diferentes.

Como bien se ha indicado, se ha cogido una a una cada dirección IP, y se ha redireccionado desde un script lanzado en Matlab que abriera una terminal Linux. La terminal que se ha utilizado en este caso es perteneciente a Linux pero utilizada en el sistema operativo de Windows. Para poder utilizar esta terminal de Linux ha sido necesario activar el modo desarrollador en Windows. Una vez que estamos en este modo, se activa en el panel de control que queremos utilizar el subsistema de Linux. Finalmente, ya está disponible en nuestro ordenador para su uso, el terminal Linux

Una vez que ya tenemos la terminal de Linux disponible, utilizamos el comando *whois* con el que direccionaremos desde el script de Matlab y buscará a que proveedor pertenece dicha dirección IP. Para no ralentizar tanto la ejecución de la búsqueda, primero se busca si la dirección IP a buscar ya había sido buscada antes, si se ha encontrado la dirección IP en la lista de las direcciones ya buscadas anteriormente se salta a la siguiente dirección IP a buscar.

Antes de continuar, es necesario conocer cómo funciona el comando *whois*, que se ha mencionado. *Whois* [29] es un comando que busca en la base de datos que almacena a los usuarios registrados de un recurso de Internet, "quién es" para obtener información sobre el propietario de un nombre de dominio en particular o una dirección IP. La información que proporciona el comando es el nombre, dirección, correo electrónico, número de teléfono y los servidores de nombre que usa el dominio. Para obtener tal información, basta con poner el siguiente comando: *whois direcciónIP*.

En la figura 3.3 se puede observar la información más relevante que devuelve este comando para la dirección IP 54.85.105.226. En ella, podemos observar el dominio al que pertenece, además se puede ver el rango de direcciones en el que se encuentra, fecha en el que el dominio fue registrado, la fecha en la que ha sido actualizado dicho dominio y un poco más de información que se puede hacer pública. En algunos casos, puede ser que no aparezca toda la información que se ha indicado, esto es debido a que los propietarios

deciden si quieren hacer pública dicha información.

```
raquel@DESKTOP-GAK2K5G:/mnt/c/Windows/System32$ whois 54.85.105.226

#
# ARIN WHOIS data and services are subject to the Terms of Use
# available at: https://www.arin.net/resources/registry/whois/tou/
#
# If you see inaccuracies in the results, please report at
# https://www.arin.net/resources/registry/whois/inaccuracy_reporting/
#
# Copyright 1997-2019, American Registry for Internet Numbers, Ltd.
#


NetRange:       54.72.0.0 - 54.95.255.255
CIDR:           54.80.0.0/12, 54.72.0.0/13
NetName:        AMAZON-2011L
NetHandle:      NET-54-72-0-0-1
Parent:         NET54 (NET-54-0-0-0-0)
NetType:        Direct Allocation
OriginAS:       AS16509
Organization:   Amazon Technologies Inc. (AT-88-Z)
RegDate:        2013-11-25
Updated:        2013-11-25
Ref:            https://rdap.arin.net/registry/ip/54.72.0.0


OrgName:        Amazon Technologies Inc.
OrgId:          AT-88-Z
Address:        410 Terry Ave N.
City:           Seattle
StateProv:      WA
PostalCode:     98109
Country:        US
RegDate:        2011-12-08
Updated:        2017-01-28
Comment:        All abuse reports MUST include:
Comment:        * src IP
Comment:        * dest IP (your IP)
Comment:        * dest port
Comment:        * Accurate date/timestamp and timezone of activity
Comment:        * Intensity/frequency (short log extracts)
Comment:        * Your contact details (phone and email) Without these we will be unable to identify
f the IP address at that point in time.
Ref:            https://rdap.arin.net/registry/entity/AT-88-Z
```

Figura 3.3: Ejemplo del comando *Whois 54.85.105.226*

Se ejecutó varias veces el comando *whois* para comprender como funcionaba, pero surgieron ciertos problemas al usar dicho comando. Como se ha podido ver en la imagen no solo devuelve únicamente el dominio al que pertenece, sino que muestra más información respecto a la que realmente necesitamos nosotros.

Una vez observados varios ejemplos del comando, el campo en el que siempre se mostraba la información necesaria es el campo *NetName*. Unas veces aparecía las letras en mayúscula y otras veces en minúscula, dato que hubo que tener en cuenta para la búsqueda del campo específico con la información requerida. En la figura 3.3, se observa que en el campo *NetName* aparece Amazon. En esa figura también se puede observar otros campos como *Organization* y *OrgName* que también señalan el nombre del operador, pero en la mayoría de los casos que se observaron no siempre aparecían estos campos.

A pesar de intentar buscar un único campo que fuera válido para la búsqueda del dominio de todas las direcciones IP que se encuentran en los archivos, no fue posible encontrarlo. En algunas de las búsquedas realizadas, el campo *NetName* no era posible encontrarlo, por tanto, en los casos en los que no se encontraba dicho campo hubo que buscar otro campo específico alternativo en el que se encontraba la información que necesitamos. Este campo, para los casos en que no se encontraba la información en *NetName* fue *Organization*.

En la figura 3.4, se puede observar cuando ejecutamos el comando *whois* aplicando el

filtro para que solo muestre el campo *netname* no muestra nada (esto mismo ocurría con NetName), pero aplicando el filtro de *Organization* si muestra esa información. En dicha imagen, se puede observar el problema que se ha mencionado respecto de los campos por los que se filtra.

```
raquel@DESKTOP-GAK2K5G:/mnt/c/Windows/System32$ whois 146.148.57.172 | grep netname
raquel@DESKTOP-GAK2K5G:/mnt/c/Windows/System32$ whois 146.148.57.172 | grep Organization
Organization:   Google LLC (GOOGL-2)
```

Figura 3.4: Se filtra por *netname* y se filtra por *Organization*

Volvió a surgir el mismo problema, no se encontraba ni el campo *NetName* en mayúsculas ni minúsculas, ni *Organization*. Por tanto, se volvió a encontrar otro campo por el que filtrar, en este caso fue *nserver*, que pertenece al servidor de la dirección IP correspondiente. Se observó que cuando fallaban los campos mencionados anteriormente se podía observar la información que necesitábamos en dicho campo. En la figura 3.5, se puede observar que del campo *nserver* se puede obtener la información necesaria de la dirección IP.

```
raquel@DESKTOP-GAK2K5G:/mnt/c/Windows/System32$ whois 177.72.240.143

% Joint Whois - whois.lacnic.net
%  This server accepts single ASN, IPv4 or IPv6 queries

% Brazilian resource: whois.registro.br


% Copyright (c) Nic.br
%  The use of the data below is only permitted as described in
%  full by the terms of use at https://registro.br/termo/en.html ,
%  being prohibited its distribution, commercialization or
%  reproduction, in particular, to use it for advertising or
%  any similar purpose.
%  2019-06-04T16:49:22-03:00

inetnum:     177.72.240.0/21
aut-num:     AS53032
abuse-c:     SPAMA4
owner:       A100 ROW SERVICOS DE DADOS BRASIL LTDA
ownerid:     12.147.176/0001-50
responsible: Tina Morris
country:     BR
owner-c:     TIMOR64
tech-c:      TIMOR64
inetrev:     177.72.240.0/21
nserver:     pdns1.ultradns.net
nsstat:      20190602 AA
nslastaa:    20190602
nserver:     x1.amazonaws.com
nsstat:      20190602 AA
nslastaa:    20190602
nserver:     x2.amazonaws.com
nsstat:      20190602 AA
nslastaa:    20190602
nserver:     x3.amazonaws.org
nsstat:      20190602 AA
nslastaa:    20190602
nserver:     x4.amazonaws.org
```

Figura 3.5: Información que se observa en el campo *nserver*

Finalmente, a la hora de buscar a que proveedor pertenece una dirección IP se decidió empezar buscando por el campo que era más genérico, *NetName*, y se ha observado que se encontraba en la mayoría de las búsquedas de direcciones IP que hemos tomado como ejemplo para analizar este caso, si este campo no existía y no se tenía ninguna información, se filtraba por el siguiente campo que habíamos observado que requería la información interesada. Así, hasta que se tenía la información deseada de la dirección IP correspondiente. Este proceso, como bien se ha indicado, se ha realizado para todas las direcciones IP que

se encuentran en los datos y se ha obtenido finalmente la información requerida para cada una de ellas. En la figura 3.6, se puede observar gráficamente lo explicado y el orden que se ha seguido para el filtro del campo.

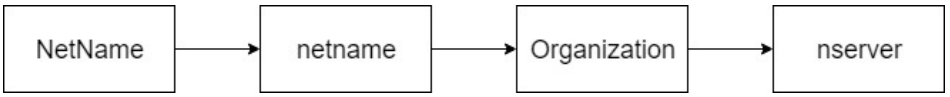


Figura 3.6: Diagrama del orden seguido para filtrar por los campos

Otro problema que ha surgido es que de algunas direcciones IP de las que se buscaron a que operador pertenecían eran privadas. Esto se debe a que su red de enrutamiento interna es asignada por espacio privado. En la figura 3.7, se puede observar un ejemplo de lo que sucede. Suponemos que nos encontramos en el host A, se desea acceder como destino al host D, para poder llegar a dicho host debe pasar por los enrutadores B y C. B y C, ambos tienen interfaces en la red pública (155.10.30.1 y 132.277.62.1). Sin embargo, su red de enrutamiento interna corresponde con una red privada. Entonces, el enrutador B puede llegar a D pasando por C. Se ha podido observar en la figura que B cuenta con una ruta que va a Internet, la red pública. Pero en la red se da una mejor ruta, es decir, hay una métrica más baja en B pasando por C. Por lo tanto, se ve la red privada en las trazas de traceroute.

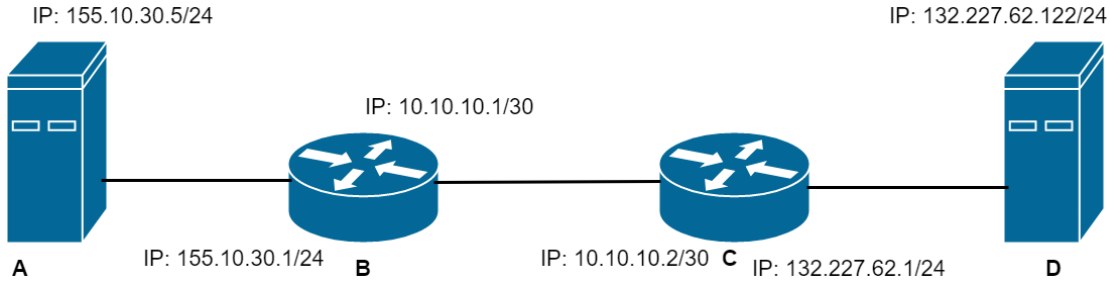


Figura 3.7: Escenario direcciones privadas en una ruta

Destacar que las direcciones IP que pertenecen a la red privada es un porcentaje muy bajo para los datos proporcionados, no llega a un 0.15 % entre todas las direcciones IP que se encuentran en los datos.

Toda esta información sobre los diferentes proveedores que se encuentran en los datos, será útil posteriormente para la elaboración de las rutas, pues facilitará su creación. A continuación, en la tabla 3.3, se puede observar la base de datos de ISPs que se han encontrado en este proceso. En ella, se observa que algunos operadores contemplan dentro de ellos a otros, esto se debe a que son subsidiarios de estos.

1	Amazon Techonologies
2	Nippon Telegraph and Telephone Corporation (NTT)
	Verio
	HKNet
3	Telecom Italia Sparkle S.p.A.
4	CenturyLink Communications
5	Telstra Global Internet Services
	Reach
6	Level 3 Communications
7	TW Telecom
8	Telefonica International Wholesale Services
9	Tinet S.p.A
10	Telia Network Services
11	Google
12	CyrusOne
	EP.NET
13	Equinix
14	TPG Telecom Limited
15	Rackspace Cloud
16	Vocus Group Limited
17	Hurricane Electric LLC
18	PTT
19	Zayo Group
	AboveNet
20	Tata Communications Limited
21	PCCW Ltd
22	LONAP
23	Verizon Communications
24	Pacnet Limited

Tabla 3.3: Base de datos de los operadores

3.5. Tabla de rutas

Una vez que ya se ha demostrado que tenemos unos datos coherentes, una lista de los posibles operadores que nos podemos encontrar cuando un paquete atraviesa la red, y sabemos cómo buscar una dirección IP de las muestras de traceroute a través del terminal de Linux mediante un script lanzado en Matlab. Entonces es posible elaborar diferentes rutas para las series temporales que se encuentran en los datos.

Creemos que es necesario organizar las rutas entre los 13 centros de datos de los diferentes CSP en estudio como una tabla, en la que verticalmente tengamos los orígenes y horizontalmente los destinos, y en cada celda de esta tabla, se podrá observar la ruta seguida en el momento temporal dado. Este conjunto de rutas en un momento temporal concreto entre todos los centros de datos en estudio forman una tabla de rutas.

De este modo el enrutamiento durante el periodo de medida quedará caracterizado por un conjunto de tablas de rutas, es decir, se tendrá una tabla para cada día y cada hora y el conjunto de todas las tablas comprenderá el enrutado con el tiempo. Aclarar que se han obviado los instantes de tiempo en los que faltaban datos en todos los archivos que

teníamos disponibles, por tanto, esos instantes no aparecerán en las tablas.

En cada una de las tablas de rutas que se forman, se podrá observar los diferentes saltos que se dan en ese instante entre un centro de datos origen y uno destino. Además, se podrá observar si los saltos que se dan se mantienen dentro de la propia infraestructura del CSP, que en ese caso funciona como un operador de tráfico, o se va a la infraestructura de otros ISP públicos.

Para la elaboración de estas tablas, ha sido necesario recorrer cada uno de los archivos de datos que contiene la información de la herramienta traceroute para cada par origen y destino. Para cada archivo, se tiene en cuenta la primera línea de cada muestra, pues como bien se ha indicado anteriormente en ella se encuentra el día y la hora de la muestra, por tanto, tendremos a que muestra o instante pertenece, es decir, sabremos a que tabla pertenece. En las siguientes líneas, se va cogiendo cada dirección IP por las que van transitando los paquetes en esa muestra y para ese origen y destino. Pero antes de buscar a quien pertenece dicha dirección IP, es necesario para evitar una alta ralentización del procesado del archivo, aplicar como un tipo de filtro en el RTT entre la dirección IP anterior y la que en este momento se quiere buscar, es decir, en cada salto que da el paquete en la red. Con ello, lo que se quiere conseguir es que dicho salto en la ruta haya salido de la infraestructura propia en la que se encontraba dicho paquete.

Para poder filtrar por un RTT adecuado, primero se ha tenido que observar el RTT en cada salto, proceso que se ha realizado manualmente observando las muestras de gran parte de los archivos proporcionados. Se observa que la mayoría de las muestras, el salto de cambio de operador en la infraestructura está entre un RTT de: 4ms a 300 ms, siendo este un rango demasiado amplio. Pero se observó que casi la mayoría cuando cambia de infraestructura era mayor de 8 ms, a excepción de algunos saltos que eran inferiores. Por tanto, se decidió filtrar por 8 ms, dado que filtrando por este valor el error cometido no era significativo.

Para cada dirección IP que pasaba este filtro del RTT, a continuación, pasaba a una función creada, donde pasaba por el comando *whois* y se filtraba la búsqueda con los campos que se han explicado antes hasta encontrar la información que necesitamos, el operador o ISP al que pertenece.

Una vez que tenemos a que ISP pertenece esa dirección IP, se ha decidido una manera de simplificar dicho nombre. En principio, se decidió abreviar cada operador con tres letras, dado que sería una manera más descriptiva del operador al que pertenece y que no hubiera confusión. Posteriormente, nos dimos cuenta que para poder comparar las tablas de rutas entre sí, y cada una de las rutas, esto es algo que dificulta la comparación, por tanto, se decidió asignar una única letra en mayúscula, dado que debido al número de operadores que se han encontrado, que se han mostrado en el apartado anterior, no supera el número de letras del alfabeto español. En la tabla 3.4, se puede observar que letra se le ha asignado a cada operador.

Después de tener a que operador pertenece la dirección IP a clasificar en ese momento, se asigna la posición de la tabla correspondiente dependiendo del centro de datos de origen y destino. Para las posiciones correspondientes de la tabla, se ha asignado en la fila el centro de datos origen al que pertenece y en la columna el centro de datos destino. Si en dicha posición ya había una letra asignada a otro operador, se concatena con el que ya había en esa celda de la tabla, pues son los diferentes saltos que hay en esa ruta concreta. Se tiene también en cuenta que si el último operador que hay en esa posición corresponde con el mismo operador de la dirección IP que se está concatenando, no se pone dado que

1	Amazon Techonologies	A
2	Nippon Telegraph and Telephone Corporation (NTT)	B
3	Telecom Italia Sparkle S.p.A.	C
4	CenturyLink Communications	D
5	Telstra Global Internet Services	E
6	Level 3 Communications	F
7	TW Telecom	G
8	Telefonica International Wholesale Services	H
9	Tinet S.p.A	I
10	Telia Network Services	J
11	Google	K
12	CyrusOne	L
13	Equinix	M
14	TPG Telecom Limited	N
15	Rackspace Cloud	O
16	Vocus Group Limited	P
17	Hurricane Electric LLC	Q
18	PTT	R
19	Zayo Group	S
20	Tata Communications Limited	T
21	PCCW Ltd	U
22	LONAP	V
23	Verizon Communications	W
24	Pacnet Limited	X

Tabla 3.4: Cada operador con su letra asignada para poder facilitar la elaboración de las tablas de rutas

en ese caso no ha salido de la infraestructura del operador y la letra a concatenar sería la misma que la última de la ruta, esto no tiene sentido dado que seguiría dentro de la red del operador. En ese caso el valor del filtro del RTT que se ha utilizado sería demasiado bajo, por eso ocurre que sigue aún dentro de la infraestructura, pero esto sucede para no cometer demasiado error, y no omitir saltos en la infraestructura. Otro de los motivos por lo que puede suceder esto es por el error experimental debido a la congestión de la red en el momento que se realizó la medida de traceroute.

Finalmente, la manera buscada para encontrar el final de la ruta es encontrar la dirección IP destino del centro de datos en la muestra de ese instante. Terminada una muestra, seguirá con la siguiente muestra así hasta terminar el archivo. Y una vez que se termina el archivo, se sigue con el siguiente archivo que tendrá otros centros de datos origen y destino distintos. Se seguirá así hasta completar las tablas de rutas y no queden más archivos que recorrer.

A continuación, en la tabla 3.5 se puede observar a modo de ejemplo como es una tabla de un instante de tiempo que contiene todas las rutas entre los centros de datos, en esta tabla se muestra una parte de la tabla total de la muestra 20. En ella, se puede observar que en la columna situada a la izquierda nos encontramos con los centros de datos de los orígenes, en la segunda fila, los centros de datos destinos en función del CSP al que pertenece, que este se puede observar en la primera línea de la tabla. Y en el interior de las celdas, se encontraría la ruta en dicho instante. En el Anexo B, se podrá observar el

ejemplo completo de la tabla de rutas en el instante 20, que para su mejor visualización ha tenido que ser dividida en dos tablas. Dentro de esta tabla se puede observar los diferentes ISP que transporta el tráfico entre ellos o si es la propia infraestructura la que atraviesan los paquetes de traceroute.

	Amazon EC2			
Origen/Destino	California	Dublin	Sao Paulo	Singapore
California		ABA	ACA	AB
Dublin	AI		AIHA	AB
Sao Paulo	AFBFBA	AHFHBHBABA		AC
Singapore	ABA	ABA	ACA	
Sydney	AED	AEIA	AEC	AB
Virginia	A	A	AJFAF	ABA
Hogn Kong	OX	OXA	OXFA	OX
London	OSFSJFSFJA	OA	OFWHFAFHA	OTA
Sydney	OP	OXA	OEC	OPM
Virginia	OA	OF	OHFSFHFCFHFAFA	OTAT
Belgica	KI	K	KC	KMLMA
Iowa	KIA	KFA	KC	KM
Taiwan	K	KF	KA	KA

Tabla 3.5: Ejemplo de Tabla de rutas

Algunos de los problemas que han surgido durante la realización de las tablas de rutas, es que en la búsqueda de una dirección IP en las líneas que devuelve el comando traceroute puede ser que nos encontremos con tres *, por tanto, incluimos que este tipo de líneas sean ignoradas y no influya en la elaboración de las tablas. También puede ocurrir que en una línea de los tres paquetes que manda traceroute, alguno de los paquetes tampoco llegue a ningún destino o no haya recibido respuesta y aparezca *. En ese caso llegamos a la conclusión que es mejor no tenerlo en cuenta.

Otro dato a tener en cuenta, es que en una muestra de la serie temporal nunca se encuentre la dirección IP destino, dado que en ese momento puede ser que no responda a traceroute y no se sabría cuando llega al final de la muestra. Por tanto, para poder detectar que se ha llegado al final de una muestra, es necesario tener en cuenta las líneas en blanco que hay entre una muestra y otra muestra en el archivo. Si no ha encontrado la dirección IP destino, y a continuación de una línea en blanco nos encontramos con la línea de la siguiente muestra que contiene el día y la hora de dicha muestra, esto significa claramente que el CSP destino no ha respondido a traceroute y hay que dar por finalizada la muestra anterior.

3.6. Métricas

Una vez que se tienen las tablas de rutas de las series temporales formadas, y con nuestro objetivo principal en mente, conocer cómo se comportan las rutas en el tiempo. Es necesario poder contrastar de alguna manera cómo varían las rutas en el tiempo. Esto se consigue comparando las diferentes tablas de rutas en cada instante de tiempo.

Para poder comparar las tablas de rutas se han utilizado diferentes métricas. En este caso son las tres métricas que se presentan a continuación:

- Acierto o fallo. Es la primera métrica que se ha utilizado. Se basa en controlar si son iguales las rutas que se están comparando o si hay algo diferente en ellas. Esto quiere decir, que si se tiene una ruta ABC y la siguiente muestra con la que se compara es ABC, se asigna un 1 porque son exactamente iguales. Si en cambio tenemos ABC y la siguiente muestra la ruta es ACB se le asigna un 0, porque no es completamente igual. Otro detalle a tener en cuenta en esta métrica es la longitud de las rutas, si ambas rutas no tienen la misma longitud se le asigna directamente un 0, dado que de primeras ya se observa que las rutas no van a ser iguales.

- Porcentaje de aciertos. Con esta métrica se podrá observar el porcentaje de aciertos que suceden a la hora de comparar dos rutas. Esto es, si coincide operador y posición en la ruta se considera acierto y por tanto, se suma 1. Si esto no sucede no se suma nada. Se compara la ruta entera y finalmente se obtiene la suma de las coincidencias encontradas entre ambas rutas, y se divide entre la longitud de la ruta del primero. Un ejemplo de esto es, si tenemos la cadena ABCDE, y la segunda cadena con la que comparamos es AECD. Tendríamos 3 letras que coinciden operador y posición, por tanto, tendríamos $3/5 = 0.6$. Con esto se observa que la ruta se parece 0.6 sobre 1.

- Distancia Levenshtein [30]. Esta métrica se basa en el número mínimo de operaciones que se necesitan para transformar una cadena de caracteres en otra. Respecto a operación se entiende la acción de insertar, eliminar o sustituir un carácter. Para dejar más clarificado cómo funciona esta distancia vemos a continuación un ejemplo. Teniendo una cadena ABCDE y como segunda cadena AFCD, tendríamos que realizar dos operaciones para convertir la segunda cadena en la primera. Se debería realizar una sustitución en la posición 2, cambiar la F por la B, y una inserción en la posición 5, en dicha posición se inserta la letra E. Con esto obtendríamos una distancia de dos. Para poder comparar con las diferentes métricas, ha sido necesario realizar un proceso de normalización, para que así todas las métricas estuvieran evaluadas de 0 a 1. Por tanto, para normalizar la distancia obtenida se divide entre la cadena más larga, y este resultado se le resta a 1, para así obtener la similitud de las cadenas y poder estar en igualdad con las demás métricas.

Estas métricas son posibles de realizar gracias a que a cada operador se le ha asignado una letra como bien se ha explicado anteriormente. Aunque en las dos primeras métricas se podían haber realizado de la manera propuesta inicialmente, tres siglas para cada operador. Pero esta manera es imposible para la tercera métrica dado que solo se puede realizar con una sigla, pues compara letra a letra cada cadena. Por tanto, para no realizar cada comparación de una manera diferente, se decidió utilizar la manera que era compatible para las tres métricas.

A continuación, se pueden observar los resultados obtenidos al aplicar cada una de las métricas explicadas, se han aplicado a las tablas de rutas que se han obtenido de las series temporales para poder compararlas. Para ello, se ha elegido un centro de datos origen a modo de ejemplo, en este caso ha sido London perteneciente a Rackspace, contra todos los centros de datos destinos disponibles en los datos.

En la figura 3.8, se observa la métrica de acierto o fallo. Se puede observa que la ruta de una muestra a otra o es exactamente igual o es completamente diferente como bien se ha explicado. Se observan poco las rutas del origen a los destinos debido a que las líneas están superpuestas, porque se obtienen los mismos valores.

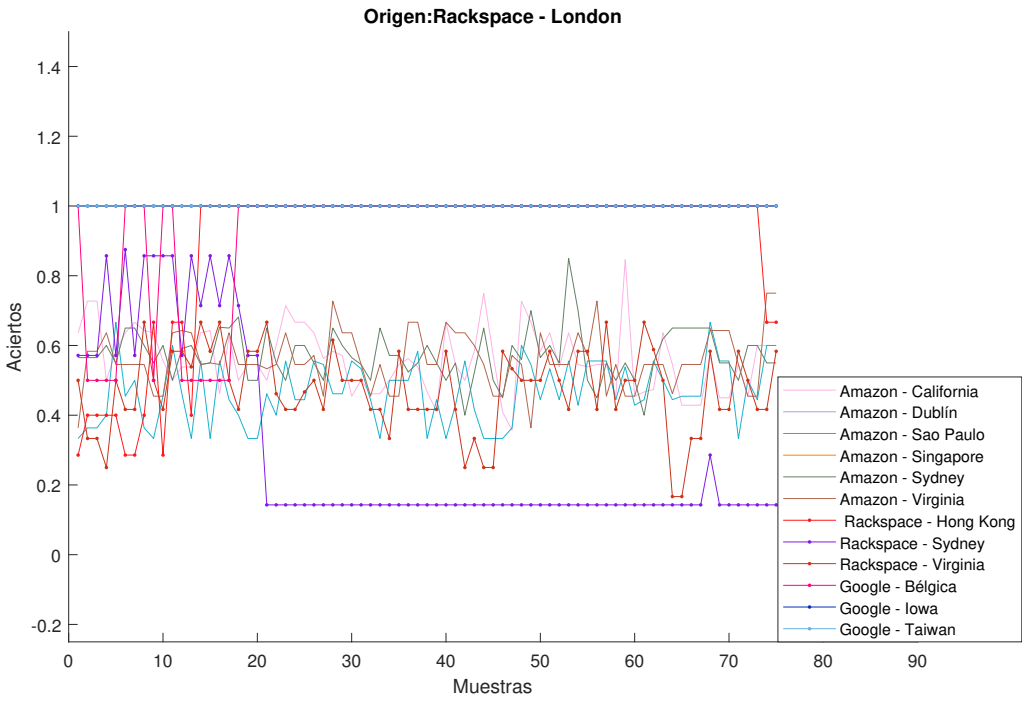


Figura 3.8: Métrica de acierto o fallo para el CSP de Rackspace con origen en London

En la figura 3.9, se observa el porcentaje de acierto, en ella ya podemos ver que hay más variabilidad en las rutas entre el origen y los destinos que en el caso anterior. Se muestra que la ruta con la que se tiene un mayor acierto es con el destino de Taiwan de Google.

En la figura 3.10, se observa la distancia Levenshtein de manera normalizada. En este caso vemos mayor parecido con la métrica anterior. Pero obteniéndose para nuestro criterio una mejor precisión con esta métrica que con la del caso anterior. Esto se puede ver en las gráficas que la mayoría de las rutas oscilan en este caso entre 0.4 a 0.6 y en el caso anterior entre 0.2 y 0.5. Con esta última métrica, se puede observar que no se contemplan rutas demasiado cambiantes, dado que más del 80 % de las rutas se encuentran por encima del 0.4.

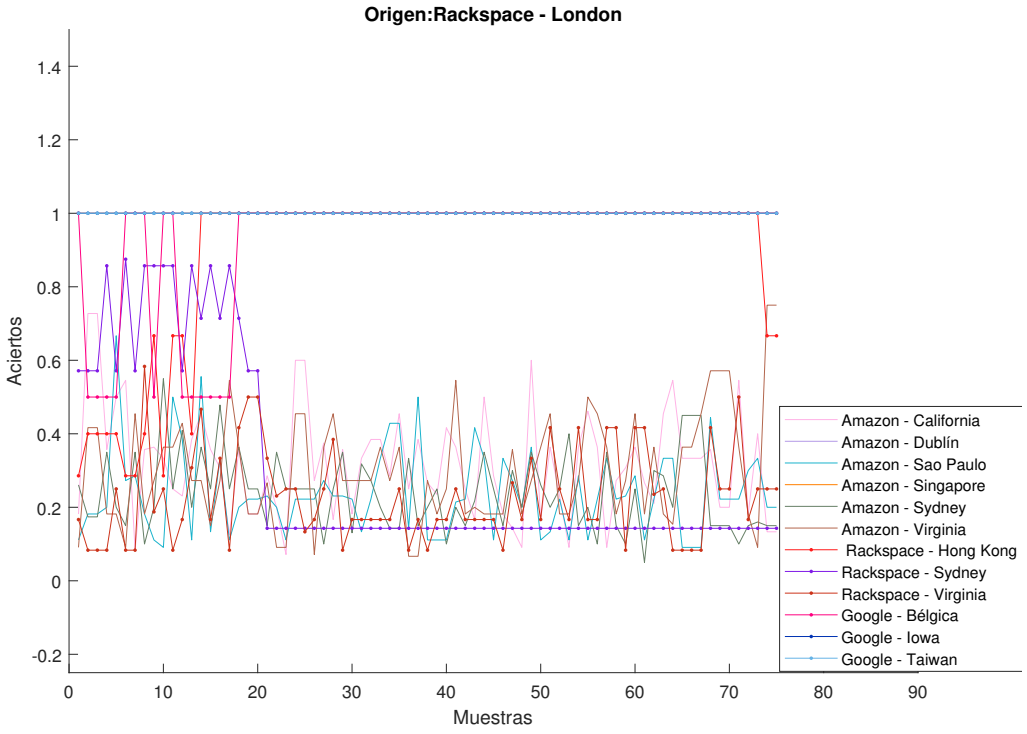


Figura 3.9: Métrica de porcentaje de acierto para el CSP de Rackspace con origen en London

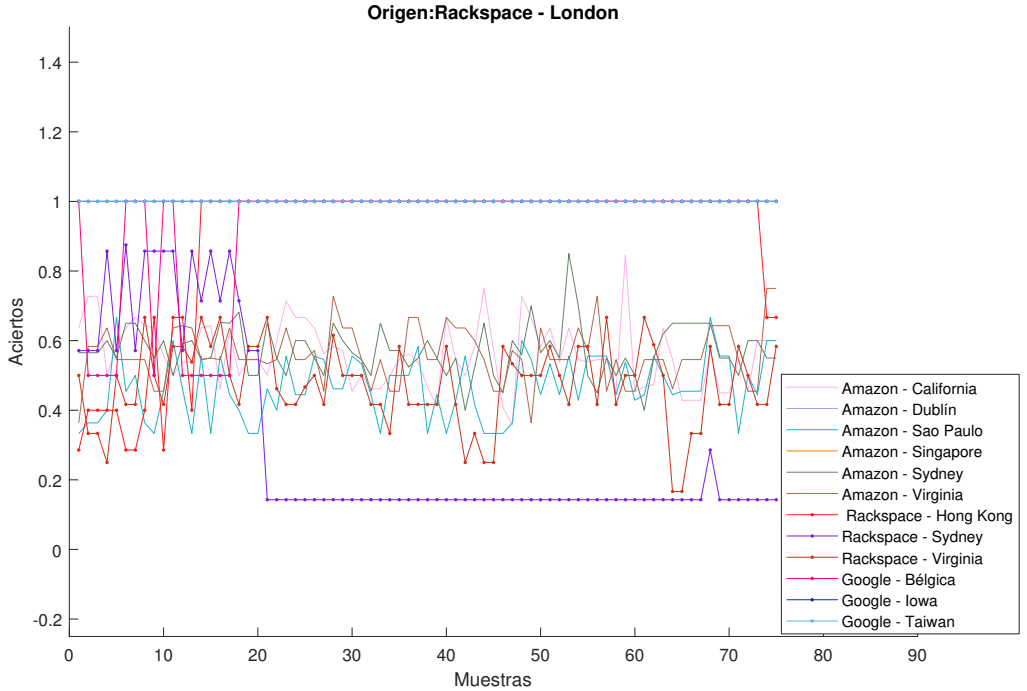


Figura 3.10: Métrica de distancia Levenshtein para el CSP de Rackspace con origen en London

3.7. Formas de comparar las tablas de rutas usando las métricas

Una vez revisado como comparar dos rutas, veamos como trasladar esto para la comparación de tablas de rutas en el tiempo, esto es, el conjunto total de rutas según avanza el tiempo. En concreto proponemos dos enfoques. El primero de ellos es comparando la primera tabla de rutas construida (esto es, la formada con la primera ejecución de traceroute) con las siguientes tablas de rutas formadas con sucesivas ejecuciones de traceroute. Se muestra en la figura 3.11, con ello se puede verificar la evolución de las rutas en el tiempo. La otra forma de comparar que hemos contemplado es observar cómo cambian las rutas de una tabla a la subsiguiente (en términos temporales) tabla de rutas, como la figura 3.12 muestra.

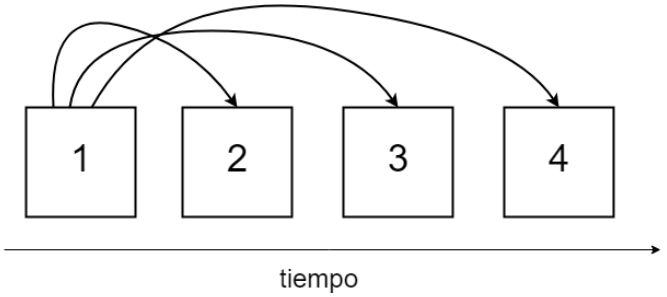


Figura 3.11: Modelo 1. Compara la primera tabla de rutas con el resto de tablas

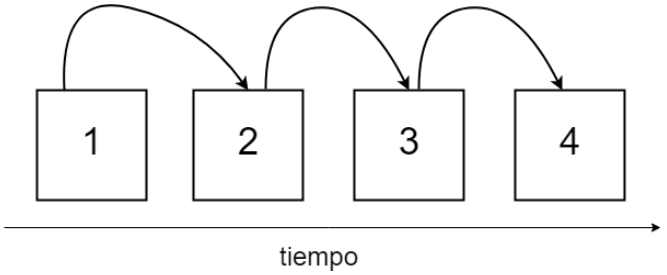


Figura 3.12: Modelo 2. Compara las tablas de rutas de una a otra

Para decantarnos por un enfoque u otro, creemos conveniente entender mejor la evolución temporal global de las tablas de rutas en el tiempo, y para ello, primero, analizar si tiene tendencia la evolución de las rutas en el tiempo (esto es, cambian entre varios estados, o con el tiempo las rutas van haciéndose cada vez más distintas de la primera ruta que vimos). Para comprobar esto, se realizan las regresiones lineales de las métricas calculadas de las tablas de rutas entre origen y destino CSP. Así, se podrá observar la tendencia que siguen las métricas. Para la realización de estas regresiones se han utilizado las funciones de Matlab: polyfit y polyval.

Como se conocen los pares de datos que se utilizan, eje x e y, se pueden determinar las incógnitas del polinomio que mejor se ajusta a una recta (regresión lineal):

$$y = ax + b$$

donde a y b son los coeficientes a calcular, x es el tiempo, es decir, los puntos en los que se quiere calcular la regresión. La y es la métrica en cada instante de tiempo.

Primero se utiliza la función polyfit, se introducen los pares de datos (x e y) y el grado del polinomio que se quiere obtener, en este caso el grado es uno. Como resultado nos devuelve los coeficientes. Una vez que hemos obtenido el resultado, con la función polyval se obtiene la regresión lineal, esto quiere decir que se evalúa la ecuación de la recta en cada uno de los puntos. En este caso, cada una de las muestras.

Destacar que en este caso las regresiones han sido realizadas para la métrica de la distancia Levenshtein. Se han realizado las regresiones para todas las rutas con la distancia mencionada. Pero a modo de ejemplo, en la figura 3.13, se muestra para el caso concreto del centro de datos de origen en Dublín del proveedor de Amazon para una mejor visualización.

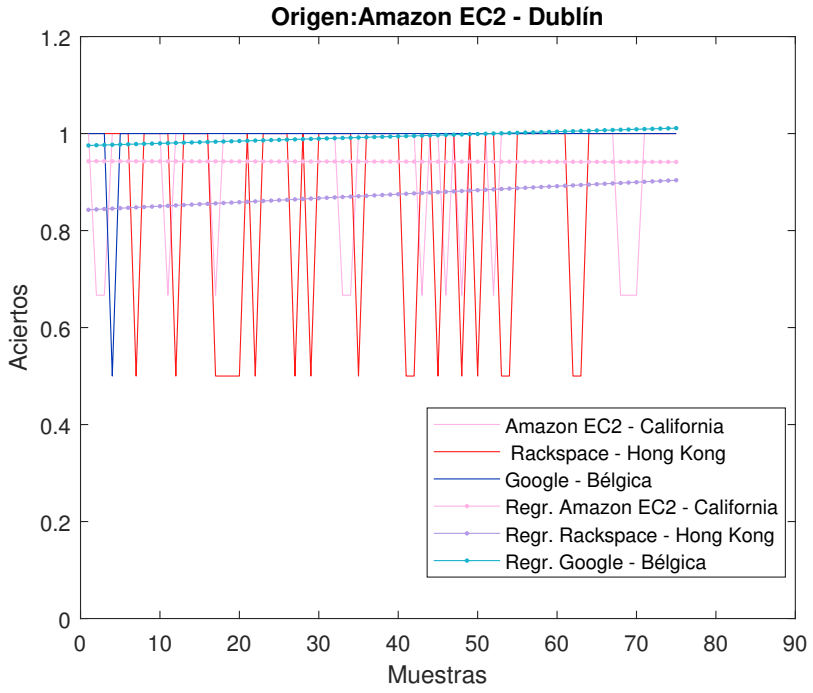


Figura 3.13: Regresión para el centro de datos origen Dublín de Amazon

En la figura 3.13, se puede llegar a observar la recta de la regresión un poco ascendente, pero ese incremento en el tiempo es mínimo. Por tanto, consideramos que no hay tendencia en las rutas y permanecen estables en la evolución del tiempo. Aunque no haya tendencia, hay que destacar que las rutas si cambian en el tiempo pero no en la evolución de este. Por tanto, a la conclusión que llegamos es que se pueden dar ciertos eventos en la red que cambien las rutas, estos eventos son aleatorios, pueden ocurrir mañana y volver al estado original. En otras palabras, pasa el evento, se actúa y cambia la ruta, pero esto no va ligado al tiempo, puede ocurrir en cualquier momento.

Al no influir el tiempo en la variación de las rutas, decidimos que para la realización de los resultados futuros se utilizará el segundo modelo propuesto. Por tanto, se observarán como cambian las rutas de un instante a otro y se analizarán los resultados.

4
Resultados

En este capítulo se mostrarán los resultados que hemos obtenido después del procesado de los datos proporcionados, en este caso las trazas de traceroute, y la comparación de las tablas de rutas las cuales se han analizado para obtener algunas conclusiones. Estos resultados que se van a mostrar a lo largo de este capítulo han sido obtenidos partiendo de lo explicado en el capítulo 3.

Para el análisis de los resultados que se van a mostrar en este capítulo, utilizaremos un análisis factorial que partirá de lo general a lo concreto. Primero analizaremos las rutas sin factores, (esto es, entendiendo todas las medidas, como medias en la nube en general); mediante el factor CSP, veremos si cada CSP se comporta distinto (esto significa que las diferencias entre las medias se explican por CSPs); como factor intermedio, veremos si hay diferencias cuando las rutas son dentro del CSP (intra) o fuera (inter); y, por último, nos centraremos en analizar las rutas por factor centro de datos origen (que es el factor más concreto).

Antes de profundizar en los resultados obtenidos, se analizará si las rutas de la nube cambian demasiado en el tiempo y en qué sentido. Si hay demasiada variación en las rutas o si por el contrario no lo hay. En nuestro caso, nos centraremos más en evaluar las rutas cambiantes, considerándolas como algo negativo que surge en la red. Consideramos que pocos cambios en las rutas se concibe como una mayor rapidez en la transmisión y rutas más estables, pero, realmente nos preguntamos si esto es así. En concreto, debemos preguntarnos, por tanto, si esto no ocasionaría rutas predecibles que conlleven a un mayor flujo de tráfico por ellas, en lugar de disponer de distintas rutas alternativas que alivien esa carga.

4.1. Nube en general

Destacar antes de entrar en detalle, que estos resultados han sido elaborados y procesados utilizando la aproximación segunda como comparativa de tablas de rutas; es decir, comparar las tablas de rutas de un día a otro, dado que la primera comparativa quedó

demostrada que no hay evolución de las rutas en el tiempo.

Para facilitar el análisis de los resultados, hemos creído conveniente realizar la distribución acumulativa de los resultados obtenidos con las métricas utilizadas, explicadas en el apartado 3.6. Estas métricas tienen un objetivo concreto, determinar si las rutas cambian en el tiempo o no. Como quedó demostrado, las rutas no se parecen más o menos por estar más/menos distantes en el tiempo en estudio. Con ello, entendemos que la parte temporal de las muestras no es nuestro foco de interés. Por tanto, las comparativas de las tablas de rutas se mostrarán como funciones de distribuciones acumulativas de portabilidad. Con este tipo de función lo que se consigue es mostrar la probabilidad acumulada para un valor dado en el eje horizontal (en nuestro caso, similitud). Por tanto, determinaremos la probabilidad de que las comparativas de las rutas sean mayor o menores a ciertos valores, es decir, se mostrarán las probabilidades de si las rutas han sufrido bastantes cambios o no.

En primer lugar, se representa la probabilidad acumulada del Cloud para las tres métricas utilizadas en el proyecto. Para ello, ha sido necesario agregar todo la información sobre la comparativa de las rutas para cada una de las métricas.

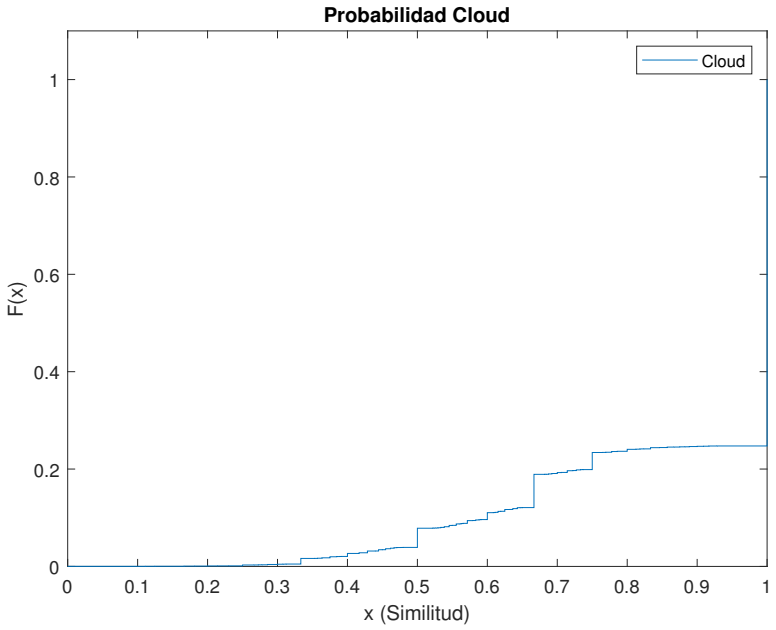
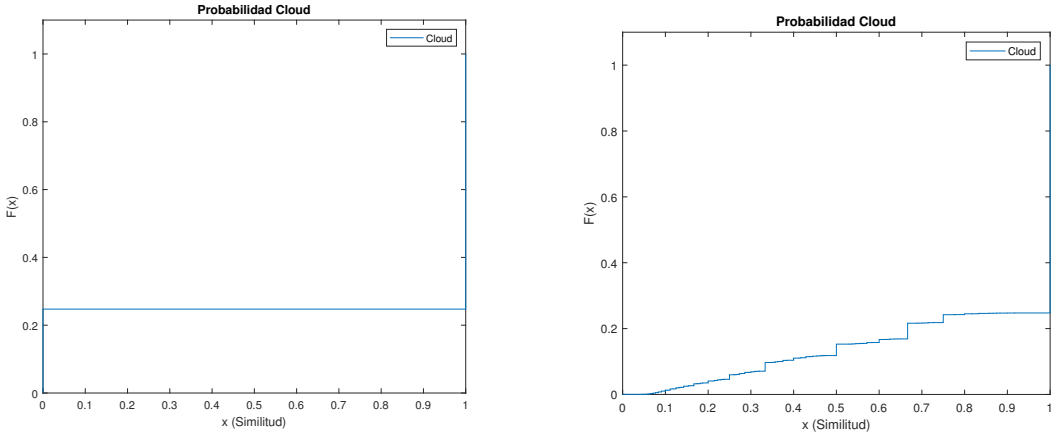


Figura 4.1: Probabilidad del Cloud en general con respecto a la similitud de las rutas con la métrica 3

En la figura 4.1, se observa la probabilidad acumulada para el caso general de la nube para la métrica 3, la distancia Levenshtein. En ella, se puede observar que entorno a un 70 % de las rutas están por encima de una similitud del 90 % y en menos del 10 % de las rutas se ha observado un mayor número de cambios, con una similitud menor o igual al 50 %. Podemos concluir, por tanto, que las rutas en rangos temporales de una semana cambian poco; prestemos más atención a las que sí varían.

En la figura 4.2(a), se muestra la probabilidad acumulativa para el caso de la métrica 1. En ella, se puede apreciar que no se distinguen grados de similitud. En concreto, el 24 % de las rutas son cambiantes. Esto se debe a la métrica utilizada de acierto o fallo.

En la figura 4.2(b), en este caso perteneciente a la métrica 2, se puede observar que



(a) Probabilidad del Cloud utilizando la métrica 1 (b) Probabilidad del Cloud utilizando la métrica 2

Figura 4.2: Probabilidad del Cloud en general con respecto a la similitud de las rutas

sigue la misma tendencia que la métrica 3. A excepción de que se muestra una mayor variación en las rutas con un 15 % de rutas cambiantes con una similitud menor del 50 %.

Para el desarrollo del resto de este capítulo, nos vamos a centrar en los resultados obtenidos con la métrica 3, distancia Levenshtein, es la que consideramos que tiene riqueza de detalle o precisión en la información que nos da. Los resultados con las métricas restantes se mostrarán en el Anexo C, en él se podrá observar que para la métrica 2 se obtienen resultados muy similares.

4.2. Los operadores en la nube

Para poder investigar de dónde provienen el 10 % de las rutas con menor similitud de la nube total, es necesario ahora separar la similitud de las rutas de la nube en general y pasarlas a las rutas por operadores. Una vez separado, investigamos de donde procede ese 10 % de las rutas cambiantes que estaba por debajo del 50 % de similitud. Ahora se muestra en la gráfica la similitud para cada operador contra todos los CSP destino, para la elaboración ha sido necesario agregar la información relacionada con las comparativas de las tablas por operadores.

En la figura 4.3, se puede observar que los operadores que presentan una peor similitud corresponden a Amazon y Rackspace. Destacando, que en la que mejor similitud se observa es el operador de Google, con un 90 % de las rutas al 100 % de similitud. En Rackspace, un 11 % de las rutas presentan menos o igual de un 50 % de similitud, contra Amazon, que presenta un 8 % de las rutas. Por tanto, nos encontramos que se obtiene peores resultados con Rackspace (si entendemos, como virtud la estabilidad de la rutas). Una de las cosas que se mencionó en la primera parte de este trabajo, es que hay que observar muy de cerca el operador Google, dado que inicialmente se obtenían pocas muestras de este operador.

Pero no nos conformamos con que operador es el que presenta las rutas más cambiantes. Nos centramos en separar las rutas intra-CSP e inter-CSP, para observar si las rutas más cambiantes se encuentran en las rutas dentro del mismo operador o en las rutas cuando cambia de operador.

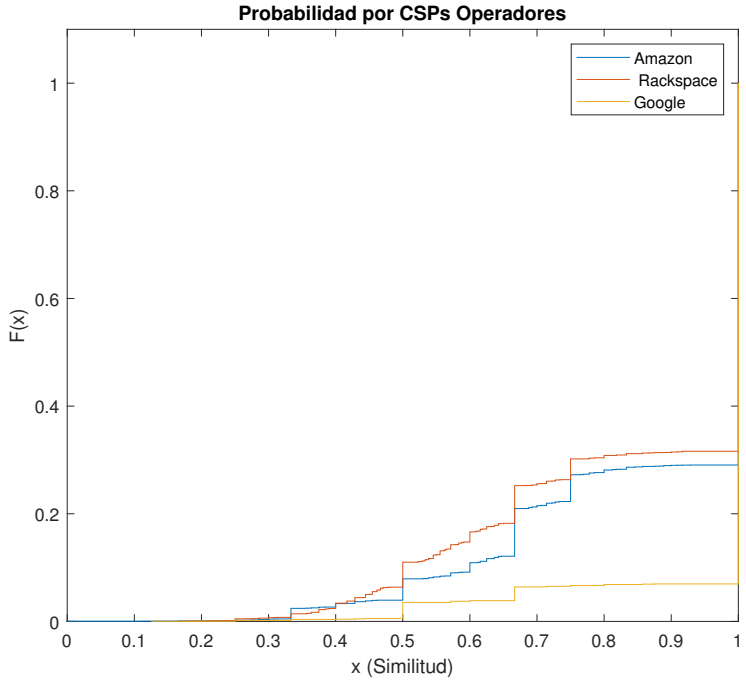


Figura 4.3: Probabilidad del Cloud por operadores con respecto a la similitud de las rutas con la métrica 3

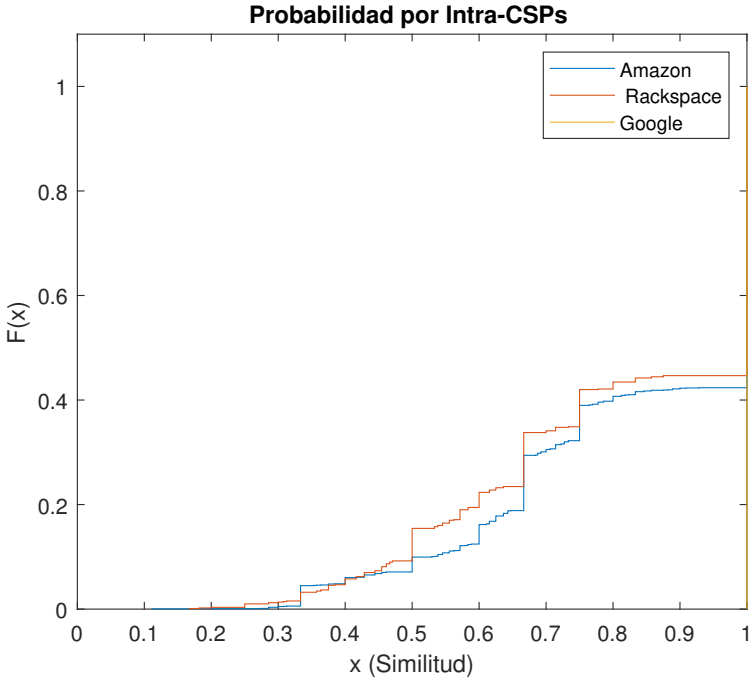


Figura 4.4: Probabilidad del Cloud por intra-CSP con respecto a la similitud de las rutas con la métrica 3

En la figura 4.4, se observa que la distribución acumulada para el operador de Google es 100 % para una similitud del 100 %, aquí nos damos cuenta que el operador Google

falla poco a la hora del enrutado dentro de la propia infraestructura, porque en este caso el número de muestras que teníamos inicialmente era similar al de los demás operadores, tema que se mostró en el apartado 3.2. En los otros dos operadores, observamos que sigue teniendo una peor similitud el operador Rackspace. Para este operador el 15 % de las rutas presentan menos de un 50 % de similitud, contra Amazon, que presenta un 10 % de las rutas. El porcentaje de las rutas para una similitud del 100 % mostrada por los operadores es, para Amazon el 58 % frente a Rackspace que muestra el 55 % de las rutas. Solo muestran una diferencia del 3 % de las rutas en el grado de mayor similitud.

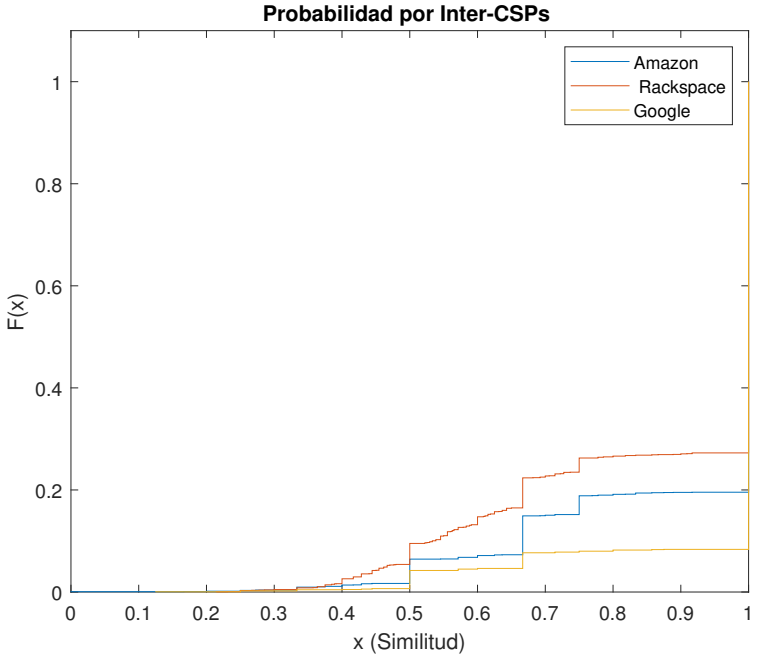


Figura 4.5: Probabilidad del Cloud por inter-CSP con respecto a la similitud de las rutas con la métrica 3

En la figura 4.5, se puede observar que en el operador de Google ahora si se dan cambios en las rutas, y no como se observó en el caso de intra-CSP. El número de muestras que se tenían para el caso de inter-CSP era escaso, pero no se ha detectado nada extraño a pesar de la falta de muestras. Se observa que tiene comportamiento mejor que los otros dos operadores, pero nada relevante porque en el caso intra-CSP, si se tenían muestras suficientes y se obtuvieron resultados muy buenos. Por tanto, puede ser que el operador de Google enrute mejor que los otros dos operadores restantes.

Respecto a los otros dos operadores encontramos que se sigue manteniendo el peor comportamiento para el operador de Rackspace, con un 73 % de las rutas con una similitud del 100 % contra Amazon con un 81 % de las rutas. Con respecto a una similitud menor o igual del 50 %, se observa que Rackspace tiene un 10 % de las rutas frente a Amazon con un 6.5 % de rutas. En este caso, también se puede observar el operador de Google con un 4 % de las rutas cambiantes.

Destacar que entre las rutas inter-CSP e intra-CSP, se ha observado que hay menos cambios en las rutas de los operadores inter-CSP. Aunque en el caso de Google, ocurre lo contrario. Se han observado mejores resultados para el caso de intra-CSP, es decir, dentro de las rutas del mismo operador.

Otro factor que creemos conveniente investigar son los centros de datos origen que originan las rutas más cambiantes, esas que nos encontramos en una similitud menor del 50 % en los operadores, que hemos podido observar en los resultados anteriores. Con este factor que queremos analizar, se podrán detectar las rutas en las que se producen un mayor número de anomalías y mayor cambios en las rutas.

4.3. Los centros de datos en la nube

Para poder investigar de dónde proceden las rutas con menor similitud, es necesario factorizar la similitud de los operadores en centros de datos origen y observar la similitud de cada uno de ellos. Una vez que son separados, investigamos de donde proceden las rutas más cambiantes que se han mostrado en el apartado anterior y que hemos creído conveniente analizar para este estudio.

En las gráficas que se muestran a continuación, se observan los centros de datos origen separados por CSP para mejorar la visualización de los resultados. Para la realización de estas gráficas ha sido necesario agregar las comparativas de las tablas de cada origen contra los diferentes destinos.

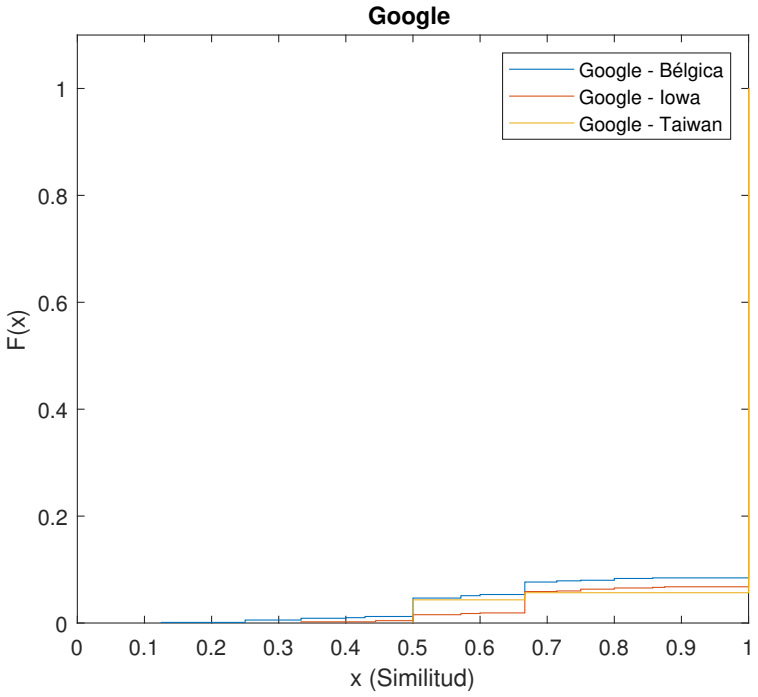


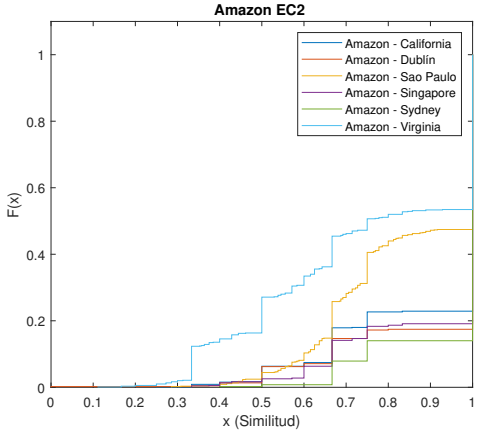
Figura 4.6: Probabilidad del Cloud por CSP origen de Google con respecto a la similitud de las rutas con la métrica 3

En la figura 4.6, se observa que para los centros de datos origen del CSP de Google, se detecta que la similitud de las tablas de rutas es peor para el centro de datos de Bélgica. Por tanto, sería el que influiría más en los malos resultados que se obtuvieron en inter-CSP en el apartado anterior.

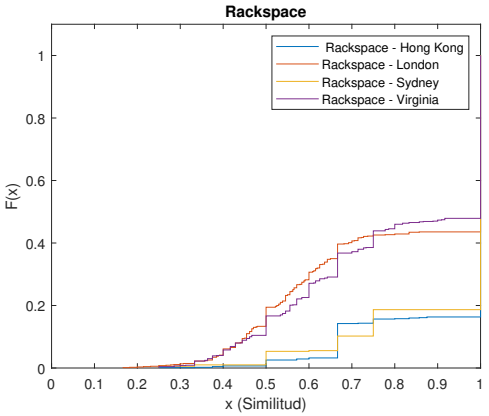
En la figura 4.7, se pueden observar los centros de datos origen para los operadores de Amazon y Rackspace. En ellas, se muestran los centros de datos origen que obtienen un

peor comportamiento en el enrutado a los diferentes destinos. Para el caso de Amazon, los centros de datos que presentan un peor comportamiento son los que tienen el origen en Virginia y Sao Paulo, con una probabilidad del 27 % y 4.5 % para una similitud menor o igual del 50 %, mientras que para una similitud del 100 % presentan una probabilidad de acierto del 47 % para Virginia y del 53 % para Sao Paulo.

Para Rackspace, los que presentan peores resultados son los que tienen origen en London y Virginia. Para una similitud del 50 % o menor se tiene una probabilidad de 19 % y 17 % respectivamente. Para una similitud del 100 %, nos encontramos con unas probabilidades del 56 % para London y un 52 % para Virginia. En ambos operadores, coincide que el centro de datos que se encuentra en Virginia obtiene malos resultados.



(a) Probabilidad del Cloud por CSP origen de Amazon



(b) Probabilidad del Cloud por CSP origen de Rackspace

Figura 4.7: Probabilidad del Cloud por CSP origen con respecto a la similitud de las rutas

A continuación, se muestra una tabla comparativa con los peores y mejores centros de datos origen que se han obtenido en los resultados para las diferentes métricas que se explicaron en el capítulo anterior. Así, se puede observar si para el resto de métricas se obtienen resultados similares o completamente diferentes a los resultados mostrados.

Métrica 1	**Mejor DC origen**	**Peor DC origen**
Amazon EC2	Sydney	Virginia
Rackspace	Hong Kong	Virginia
Google	Taiwan	Belgica

Tabla 4.1: Comparativa DC origen para cada CSP métrica 1

Métrica 2	**Mejor DC origen**	**Peor DC origen**
Amazon EC2	Sydney	Virginia
Rackspace	Hong Kong	London
Google	Iowa	Belgica

Tabla 4.2: Comparativa DC origen para cada CSP métrica 2

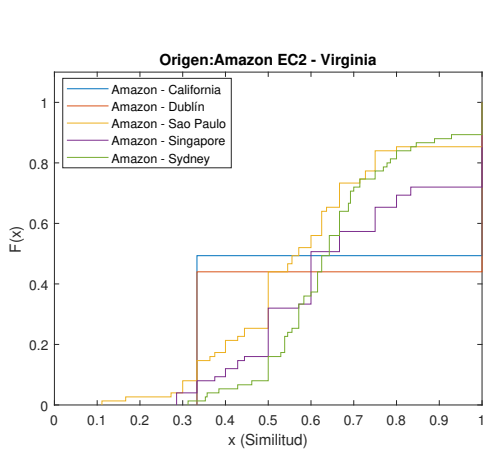
Comparando las tablas 4.1, 4.2 y 4.3, se puede observar que para algunos casos los mejores y peores centros de datos origen hay unanimidad. Esto queda demostrado para el operador de Amazon, el mejor y peor centro de datos que enrutan es el mismo para las tres

Métrica 3	**Mejor DC origen**	**Peor DC origen**
Amazon EC2	Sydney	Virginia
Rackspace	Hong Kong	London
Google	Iowa	Belgica

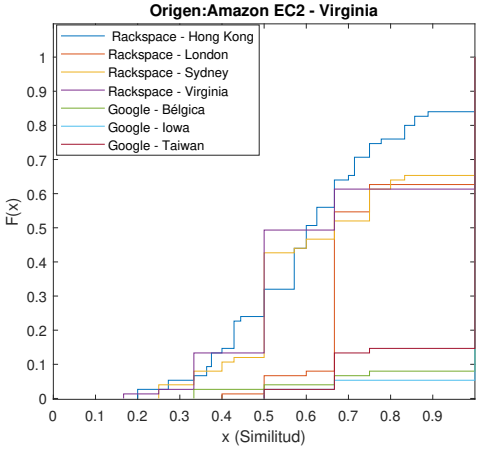
Tabla 4.3: Comparativa DC origen para cada CSP métrica 3

métricas, Sydney y Virginia respectivamente. Para el operador de Rackspace, la unanimidad se da solo para el mejor centro de datos, que es el que se encuentra en Hong Kong. Pero en el caso del peor centro de datos se encuentra en London y Virginia dependiendo la métrica utilizada. Para el operador de Google, queda demostrado que el peor centro de datos que se observa para las tres métricas, es el que se encuentra en Bélgica y los mejores centros de datos que enrutan son dos: el de Iowa y el de Taiwan.

Una vez que se han observado los resultados obtenidos para los DC origen con las diferentes métricas, ha quedado demostrado que se obtienen resultados similares entre los peores y mejores orígenes. Además, antes de finalizar este capítulo creemos conveniente analizar otro factor que es investigar más sobre los DC en los que se han obtenido peores resultados. Para ello, creemos adecuado mostrar las rutas entre el DC origen con los diferentes DC destinos, por si alguna ruta en concreto es la que genera los malos resultados en el enrutado.



(a) Probabilidad del Cloud por DC origen contra destinos Amazon



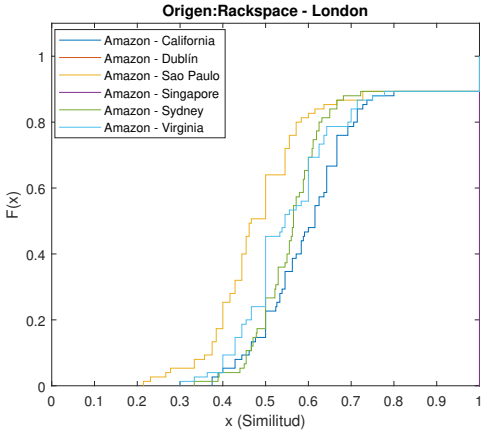
(b) Probabilidad del Cloud por DC origen contra destinos Rackspace y Google

Figura 4.8: Probabilidad del Cloud por DC origen Virginia con respecto a la similitud de las rutas para cada destino

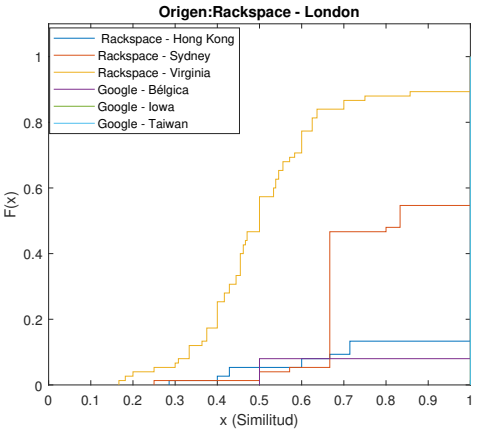
En la figura 4.8, se pueden observar que las peores rutas que se encuentra con origen en el centro de datos de Virginia son con los destinos de California perteneciente a Amazon y Virginia de Rackspace, ambos tienen una probabilidad del 49 % de rutas cambiantes para una similitud menor del 50 %. Cuando nos acercamos a una similitud del 100 %, tenemos que California tiene una probabilidad de que el 51 % de las rutas no cambien, frente a Virginia que tiene una probabilidad del 37 %. Con esto queda demostrado que la peor ruta y la que más está influyendo en las rutas cambiantes de la nube en general es la ruta entre centro de datos Virginia de Amazon con destino a Virginia de Rackspace.

A continuación, se mostrarán los dos peores centros de datos de Rackspace ya que

consideramos que los centros de datos origen de Google no influyen lo suficiente en los malos resultados obtenidos de las rutas cambiantes y creemos que son más relevantes los pertenecientes al operador de Rackspace. Dado que ha quedado demostrado que es el peor CSP en cuanto a enrutado.



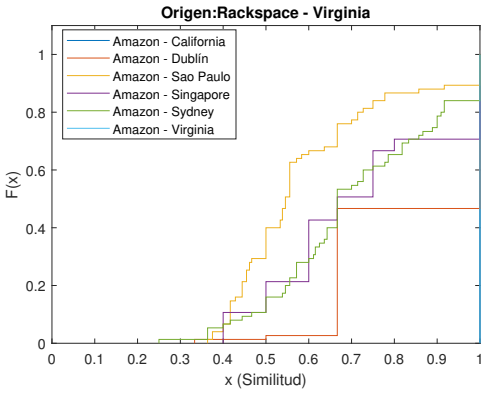
(a) Probabilidad del Cloud por DC origen contra destinos Amazon



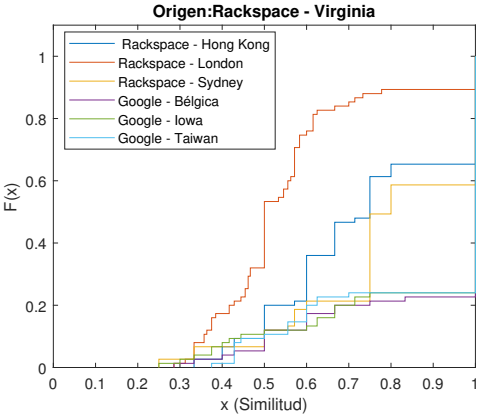
(b) Probabilidad del Cloud por DC origen contra destinos Rackspace y Google

Figura 4.9: Probabilidad del Cloud por DC origen London con respecto a la similitud de las rutas para cada destino

Empezamos mostrando los resultados del centros de datos origen de London perteneciente a Rackspace. En las figura 4.9, se observa que las peores rutas que se encuentra son los destinos de Sao Paulo perteneciente a Amazon y Virginia de Rackspace con una probabilidad del 64 % y 57 % respectivamente, para una similitud menor del 50 %. Cuando nos acercamos a una similitud del 100 %, tenemos que ambos destinos de centros de datos tienen una probabilidad similar de que las rutas no cambien. Con estos resultados, decidimos que la peor ruta y la que más está influyendo en las rutas cambiantes de la nube en general es la ruta entre centro de datos London de Rackspace con destino a Sao Paulo de Amazon.



(a) Probabilidad del Cloud por DC origen contra destinos Amazon



(b) Probabilidad del Cloud por DC origen contra destinos Rackspace y Google

Figura 4.10: Probabilidad del Cloud por DC origen Virginia con respecto a la similitud de las rutas para cada destino

El segundo centro de datos origen de Rackspace es el ubicado en Virginia. En la figura 4.10, se observa que las peores rutas se encuentran con los centros de datos destinos de Sao Paulo y London, con una probabilidad de que las rutas cambien del 40 % y 53 % respectivamente, para una similitud del 50 %. Igual que en el caso anterior, ocurre que la probabilidad es muy similar para que las rutas no cambien. Por tanto, queda demostrado que la peor ruta y la que más puede influir en los resultados anteriores es la que tiene como destino el centro de datos de London.

Métrica 3	**DC origen**	**DC destino**
Origen: Amazon EC2	Virginia	Virginia (Rackspace)
Origen: Rackspace	London	Sao Paulo (Amazon)
	Virginia	London (Rackspace)

Tabla 4.4: Peores rutas entre DC utilizando métrica 3

En la tabla 4.4, se resume lo mostrado en las figuras anteriores, se puede observar que entre los centros de datos peores tanto en origen como en destino, siempre se encuentra casi los mismos centros de datos. Además, destacar que aunque el centro de datos de Virginia del operador de Amazon solo se encuentre en este caso en el origen, es uno de los más influyentes en el mal enrutado como centro de datos destino, esto mismo ocurre con el centro de datos de Sao Paulo pero a la inversa.

Creemos conveniente, al igual que se realizó para los orígenes de los centros de datos, mostrar los resultados con las otras dos métricas, pero en este caso con las peores rutas que se han dado. Para así, poder observar si se obtiene concordancia con los resultados de la métrica 3, aunque se utilicen diferentes métricas.

A continuación, se podrán observar las tablas para las otras dos métricas restantes. Como bien se indicó anteriormente, el operador Google no creemos conveniente incluirlo en este tipo de rutas, dado que su influencia en los malos resultados en el enrutado se ha considerado mínima. Las pruebas de esto se pudieron ver con anterioridad. Por tanto, mostramos las dos peores rutas para el operador de Rackspace como se ha realizado para la métrica 3.

Métrica 1	**DC origen**	**DC destino**
Origen: Amazon EC2	Virginia	Sydney (Amazon)
Origen: Rackspace	Virginia	Virginia (Amazon) / Virginia (Rackspace)
	Virginia	Sao Paulo (Amazon) / London (Rackspace)

Tabla 4.5: Peores rutas entre DC utilizando métrica 1

Métrica 2	**DC origen**	**DC destino**
Origen: Amazon EC2	Virginia	Sydney (Amazon)
Origen: Rackspace	London	Sydney (Amazon)
	Virginia	London (Rackspace)

Tabla 4.6: Peores rutas entre DC utilizando métrica 2

En las tablas 4.5 y 4.6, se ha podido observar que no hay unanimidad con la tabla anterior respecto la métrica 3. En ellas, se puede observar que se encuentra rutas de las mostradas anteriormente, pero en este caso se incluye otro centro de datos destino, Sydney del operador de Amazon, que no había sido antes incluido en las peores rutas.

En la tabla para el caso de la métrica 1, se puede observar que hay empates en las peores rutas que nos podemos encontrar para el caso de Rackspace como origen, dado que no hay demasiada precisión en los resultados que se han observado. Respecto a la tabla correspondiente a la métrica 2, se muestran resultados que se han observado para las métricas anteriores. En ésta, si encontramos un destino único pero el siguiente peor centro de datos se encuentra muy de cerca del destacado como destino de la peor ruta.

Finalmente, destacar que en estas últimas tablas quedan reflejadas las peores rutas que se han podido observar durante este estudio, aunque no son las únicas rutas influyentes en los resultados por operadores o en la nube en general, dado que únicamente hemos mostrado una única ruta o dos para el caso de Rackspace, pero se podía hacer un análisis de un mayor número de rutas influyentes.

5

Conclusiones y trabajo futuro

En este capítulo se van a tratar los principales resultados y contribuciones de este estudio realizado sobre el enrutado en la nube pública. Al final del capítulo, se mostrarán los posibles trabajos futuros que pueden seguir las líneas desarrolladas en este Trabajo Fin de Máster.

Durante el desarrollo del estudio se han ido observando algunas conclusiones sobre el enrutado en la nube pública con el que hemos pretendido analizar la variación de las rutas durante 5 días de los proveedores predominantes en la nube, a través de los resultados obtenidos mediante la herramienta traceroute.

Cabe destacar que las rutas no varían indefinidamente en el tiempo, sino que transita entre estados con retornos a estados anteriores. Lo que puede suceder en la variación del enrutado en la nube, es que influyan los eventos que suceden en la red en un instante, es decir, puede darse que en el momento determinado en el que va a pasar el paquete por la red, ésta se encuentre más congestionada, y por tanto, la ruta que va a seguir el paquete no va a ser la misma que en momento anterior. Pero, en los instantes posteriores, puede suceder que ese paquete si siga la misma ruta que el caso anterior, por tanto, creemos conveniente destacar que no se puede seguir un patrón determinado en el enrutado en la nube pública, dado que es aleatorio dependiente de los eventos que se den en la red.

Se aplicó un análisis factorial del banco de pruebas de traceroute entre los centros de datos, con esto se concluye que las variaciones en el enrutado de la nube solo influyen un 10 %, por tanto, determinamos una notable estabilidad en las rutas. De ese porcentaje de variación, los operadores que más influyen son Amazon y Rackspace.

Respecto a los peores operadores encontrados en el enrutado, destacar que influyen más en estos malos resultados los pertenecientes a la red intra-CSP, a pesar de que se encuentren dentro de la misma red, esto puede suceder porque en ciertos casos tiene que salir fuera de esa red para poder llegar al CSP destino.

Además, destacar que se analizaron los peores DC origen que enrutan de ambos operadores. Se concluye que siempre se encuentra que el peor centro de datos origen ubicado en Virginia sin importar al proveedor de servicio al que pertenezca. Tal vez, se ve influenciado

por el punto geográfico en el que se encuentra.

Finalmente, se concluye que en el enrutado en la nube se ha observado una gran estabilidad, sin contener demasiadas variaciones en ellas. Pero esto puede suponer una insensibilidad a problemas de rendimiento.

En este estudio, las rutas no cambian lo que sugiere estabilidad, y esto indicaría rendimientos no variables y predecibles en el servicio, o lo que es lo mismo, un elemento favorable para proveedores y clientes. Ahora bien, podemos también entender que las rutas no cambian a pesar de que hayan cambiado las condiciones, esto quiere decir, que alguna ruta puede haber estado funcionando mal y los operadores no han sido capaces de reaccionar a ésto modificándolas. Por tanto, para poder comprobar ésto, deberíamos correlar los cambios en las rutas con medidas de calidad de servicio como el ancho de banda o el RTT. Esto es justo, las líneas en las que deberíamos continuar el trabajo haciéndolo más completo.

Glosario de acrónimos

- **AS**: Autonomous System
- **CSP**: Cloud Service Provider
- **DC**: Data Center
- **PDA**: Personal Digital Assistant
- **IaaS**: Infrastructure as a Service
- **PaaS**: Platform as a Service
- **SaaS**: Software as a Service
- **EC2**: Elastic Compute Cloud
- **S3**: Simple Storage Service
- **ERP**: Enterprise Resource Planning
- **CRM**: Customer Relationship Management
- **CPU**: Central Processing Unit
- **RTT**: Round-trip time
- **VM**: Virtual Machine
- **TCP**: Transmission Control Protocol
- **UDP**: User Datagram Protocol
- **PoP**: Point of Presence
- **TTL**: Time To Live
- **ICMP**: Internet Control Message Protocol

Bibliografía

[1] Markcraft Solutions. Cloud computing. https://markcraftsolutions.com/cloud-computing-training/.

[2] UniPrint. 7 different types of cloud computing structures. https://www.uniprint.net/en/7-types-cloud-computing-structures/.

[3] Cisco. Cisco global cloud index: Forecast and methodology, 2016–2021. https://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/white-paper-c11-738085.html.

[4] José Luis García-Dorado. Bandwidth measurements within the cloud: Characterizing regular behaviors and correlating downtimes. *ACM Transactions on Internet Technology*, 17(4), 2017.

[5] Giuseppe Aceto, Alessio Botta, Walter de Donato, and Antonio Pescapé. Cloud monitoring: A survey. *Computer Networks*, 57(9):2093 – 2115, 2013.

[6] Philipp Leitner and Jürgen Cito. Patterns in the chaos-a study of performance variation and predictability in public IaaS clouds. *ACM Transactions on Internet Technology*, 16(3):15:1–15:23, 2016.

[7] Valerio Persico, Pietro Marchetta, Alessio Botta, and Antonio Pescapé. Measuring network throughput in the cloud: The case of amazon EC2. *Computer Networks*, 93(3):408–422, 2015.

[8] Bradley Huffaker, Marina Fomenkov, Daniel J. Plummer, David Moore, and K Claffy. Distance metrics in the Internet. In *IEEE International Telecommunications Symposium*, pages 200–205, 2002.

[9] Sally Floyd and Vern Paxson. Difficulties in simulating the Internet. *IEEE/ACM Transaction on Networking*, 9(4):392–403, 2001.

[10] IBM. Cloud computing: A complete guide. https://www.ibm.com/cloud/learn/cloud-computing.

[11] MakeSoft Technologies. Breve historia del cloud computing. https://www.makesoft.es/es/breve-historia-del-cloud-computing/, 2016.

[12] Priyanshu Srivastava and Rizwan Khan. A review paper on cloud computing. *International Journals of Advanced Research in Computer Science and Software Engineering*, pages 17–20, 2018.

[13] Haibo Yang and Mary Tate. A descriptive literature review and classification of cloud computing research. *Communications of the Association for Information Systems*, 31, 2012.

[14] Aaqib Rashid and Amit Chaturvedi. Cloud computing characteristics and services: A brief review. *International Journal of Computer Sciences and Engineering*, 7:421–426, 2019.

[15] M. Rajendra Prasad, Ramavathu Lakshman Naik, and V. Bapuji. Cloud computing: Research issues and implications. *International Journal of Cloud Computing and Services Science*, 2:134–140, 2013.

[16] Timothy Grance Peter Mell. The NIST definition of cloud computing. *Recommendations of the National Institute of Standards and Technology*, 2011.

[17] Eric Simmon. Evaluation of cloud computing services based on NIST. *Recommendations of the National Institute of Standards and Technology*, 2018.

[18] Sujata Banerjee Sujoy Basu Sung-Ju Lee, Puneet Sharma and Rodrigo Fonseca. Measuring bandwidth between PlanetLab nodes. *Passive and Active Network Measurement Conference*, pages 292–305, 2005.

[19] Edward Walker. Benchmarking amazon EC2 for high-performance scientific computing. *Login: The Magazine of USENIX SAGE*, pages 18–23, 2008.

[20] Ignacio Bermudez, Stefano Traverso, Marco Mellia, and Maurizio Munafò. Exploring the cloud from passive measurements: The Amazon AWS case. *IEEE International Conference on Computer Communications*, pages 230–234, 2013.

[21] Srikanth Kandula Ang Li, Xiaowei Yang and Ming Zhang. CloudCmp: Comparing public cloud providers. *ACM Conference on Internet Measurement*, pages 1–14, 2010.

[22] José Luis García-Dorado and Sanjay G. Rao. Cost-aware multi data-center bulk transfers in the cloud from a customer-side perspective. *IEEE Transactions on Cloud Computing*, 7(1):34–47, 2019.

[23] Alessio Botta Valerio Persico, Pietro Marchetta and Antonio Pescapé. On network throughput variability in Microsoft Azure cloud. *IEEE Global Communications Conference*, pages 1–6, 2015.

[24] Yiyang Chang T. S. Eugene Ng Mohammad Hajjat, Ruiqi Liu and Sanjay G. Rao. Application-specific configuration selection in the cloud: impact of provider policy and potential of systematic testing. *IEEE International Conference on Computer Communications*, pages 873–881, 2015.

[25] Alessio Botta Valerio Persico, Pietro Marchetta and Antonio Pescapé. A first look at public-cloud inter-datacenter network performance. *IEEE Global Communications Conference*, pages 1–7, 2016.

[26] Pietro Marchetta Antonio Montieri Valerio Persico, Alessio Botta and Antonio Pescapé. On the performance of the wide-area networks interconnecting public-cloud datacenters around the globe. *Computer Networks*, pages 67–83, 2017.

[27] Linux man page. Traceroute. `https://linux.die.net/man/8/traceroute`.

[28] Microsoft. What is Azure firewall? `https://docs.microsoft.com/en-us/azure/firewall/overview`.

[29] LinuxConfig. Look up website information with whois in Linux. `https://linuxconfig.org/look-up-website-information-with-whois-in-linux`.

[30] Algoritmos Similaridad y Distancia. Distancia de Levenshtein. `https://sites.google.com/site/algoritmossimilaridaddistancia/distancia-de-levenshtein`.

A

Archivo con trazas de Traceroute

```
20140617154145,1403037705,Tue Jun 17 15:41:45 2014
traceroute to 119.9.88.108 (119.9.88.108), 30 hops max, 60 byte packets
 1  216.182.236.114 (216.182.236.114)  0.764 ms  0.770 ms  0.746 ms
 2  72.21.222.18 (72.21.222.18)  1.868 ms  1.912 ms  1.867 ms
 3  205.251.229.15 (205.251.229.15)  2.004 ms  2.321 ms  2.220 ms
 4  pacnet-pao1.paix.net (198.32.176.72)  3.379 ms  3.276 ms  3.301 ms
 5  be1.gw3.sjc1.asianetcom.net (202.147.50.185)  157.725 ms  157.737 ms  157.718 ms
 6  ip-61-14-158-46.asianetcom.net (61.14.158.46)  159.099 ms  157.779 ms  161.467 ms
 7  gi15-0-0.gw5.hkg3.asianetcom.net (61.14.157.138)  215.597 ms * *
 8  ge-0-1-2-0.cr4.hkg3.asianetcom.net (202.147.16.246)  158.538 ms  158.583 ms  158.514 ms
 9  gi9-0-0.gw2.hkg3.asianetcom.net (202.147.16.94)  158.349 ms  158.142 ms  158.287 ms
10  RHI-0001.gw2.hkg3.asianetcom.net (203.192.178.66)  158.654 ms  180.083 ms  179.722 ms
11  vl901.core1a.hkg1.rackspace.net (120.136.47.12)  192.438 ms  191.247 ms  191.351 ms
12  119.9.64.69 (119.9.64.69)  159.266 ms  159.273 ms  159.404 ms
13  119.9.88.108 (119.9.88.108)  158.980 ms  158.892 ms  158.590 ms

20140617164311,1403041391,Tue Jun 17 16:43:11 2014
traceroute to 119.9.88.108 (119.9.88.108), 30 hops max, 60 byte packets
 1  216.182.236.114 (216.182.236.114)  0.652 ms  0.655 ms  0.621 ms
 2  72.21.222.18 (72.21.222.18)  2.000 ms  18.373 ms  1.694 ms
 3  205.251.229.15 (205.251.229.15)  2.220 ms  2.193 ms  2.124 ms
 4  pacnet-pao1.paix.net (198.32.176.72)  3.315 ms  3.332 ms  3.315 ms
 5  be1.gw3.sjc1.asianetcom.net (202.147.50.185)  157.589 ms  157.509 ms  157.723 ms
 6  ip-61-14-158-46.asianetcom.net (61.14.158.46)  158.368 ms  160.991 ms  161.151 ms
 7  gi15-0-0.gw5.hkg3.asianetcom.net (61.14.157.138)  158.233 ms  158.202 ms *
 8  ge-0-1-2-0.cr4.hkg3.asianetcom.net (202.147.16.246)  158.199 ms  158.230 ms  158.194 ms
 9  gi9-0-0.gw2.hkg3.asianetcom.net (202.147.16.94)  158.332 ms  158.210 ms  158.235 ms
10  RHI-0001.gw2.hkg3.asianetcom.net (203.192.178.66)  158.391 ms  158.466 ms  158.317 ms
11  vl901.core1a.hkg1.rackspace.net (120.136.47.12)  159.137 ms  158.810 ms  158.904 ms
12  119.9.64.69 (119.9.64.69)  159.164 ms  159.047 ms  159.230 ms
13  119.9.88.108 (119.9.88.108)  159.236 ms  158.356 ms  158.573 ms
```

Figura A.1: Ejemplo de archivo de trazas de Traceroute que se ha utilizado para el estudio

B

Tabla de rutas de la muestra 20

Tabla B.1: Tabla de rutas

	Amazon EC2					
Origen/Destino	California	Dublin	Sao Paulo	Singapore	Sydney	Virginia
California		ABA	ACA	AB	AEA	A
Dublin	AI		AIHA	AB	AFB	A
Sao Paulo	AFBFBA	AHFHBHBABA		AC	AFHEHABABABAB	AHFHFHFBA
Singapore	ABA	ABA	ACA		ABA	ABA
Sydney	AED	AEIA	AEC	AB		AEDA
Virginia	A	A	AJFAF	ABA	ABDBEBEABAB	
Hogn Kong	OX	OXA	OXFA	OX	OE	OXA
London	OSFSJFSFJA	OA	OFWHFAFHA	OTA	OTBEBEAEBAEABAEB	OSFJFSF
Sydney	OP	OXA	OEC	OPM	O	OXM
Virginia	OA	OF	OHFSFHFCFHFAFA	OTAT	OSFEAEA	O
Belgica	KI	K	KC	KMLMA	KEAEA	K
Iowa	KIA	KFA	KC	KM	KEA	KD
Taiwan	K	KF	KA	KA	KA	KD

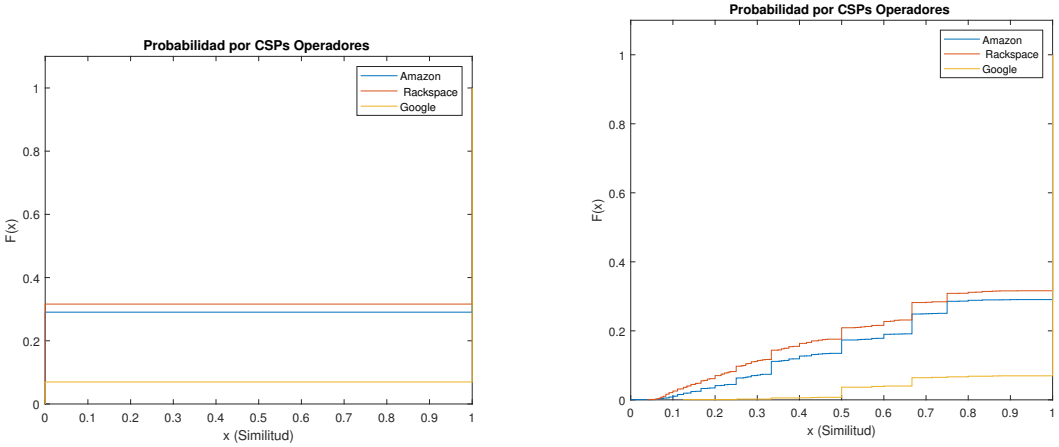
	Rackspace				Google		
Origen/Destino	Hong Kong	London	Sydney	Virginia	Belgica	Iowa	Taiwan
California	AX	AS	AP	A	AK	AK	AK
Dublin	XO	A	AO	AJO	AK	AK	AK
Sao Paulo	AFUO	AHTOTO	ACEO	AJTO	AK	AK	AK
Singapore	AX	ABF	AX	ABFO	AK	AK	AK
Sydney	AX	ABF	A	AETO	AK	AK	AK
Virginia	AMAXMAMX	AS	AN	A	AK	AK	AK
Hogn Kong		OBFBFO	OE	OTO	OK	OK	OK
London	OXO		OTEO	OJSFTSTFOFT	OK	OK	OK
Sydney	OE	OT		OPQ	OK	OK	OK
Virginia	OTOTO	OJSFJFSJFSFTF	OMQP		OSFJSFSK	OFJSK	OSJSJKFSK
Belgica	KTO	K	KEO	KS		K	K
Iowa	KX	K	KEO	KF	K		K
Taiwan	KOU	K	KE	KE	K	K	

Tabla B.2: Tabla de rutas

C

Resultados con métricas 1 y 2

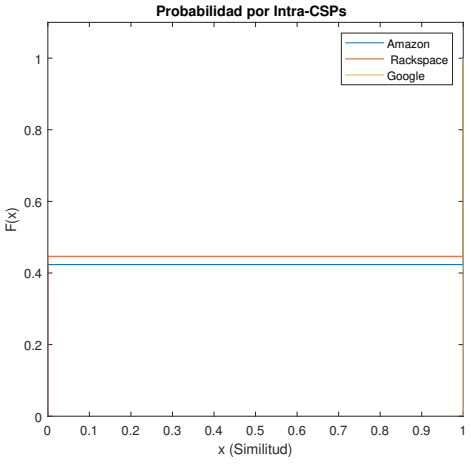
Los operadores en la nube



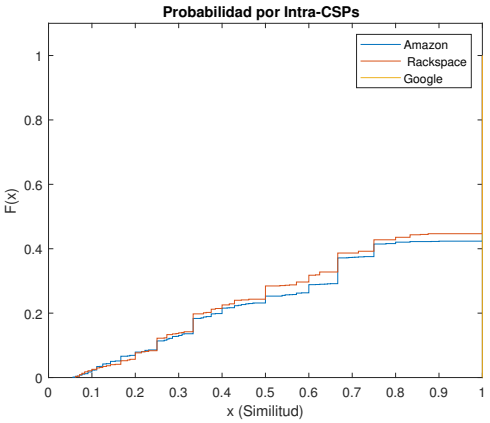
(a) Probabilidad del Cloud por operadores con la métrica 1

(b) Probabilidad del Cloud por operadores con la métrica 2

Figura C.1: Probabilidad del Cloud por operadores con respecto a la similitud de las rutas

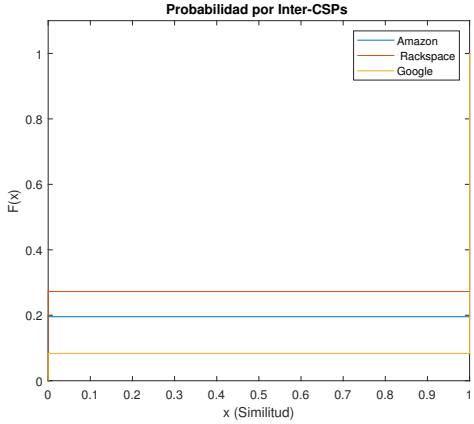


(a) Probabilidad del Cloud por intra-CSP con la métrica 1

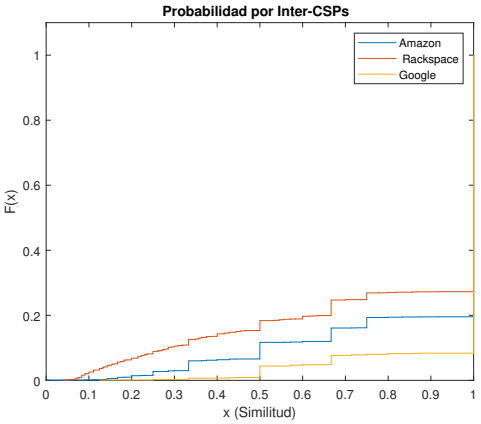


(b) Probabilidad del Cloud por intra-CSP con la métrica 2

Figura C.2: Probabilidad del Cloud por intra-CSP con respecto a la similitud de las rutas



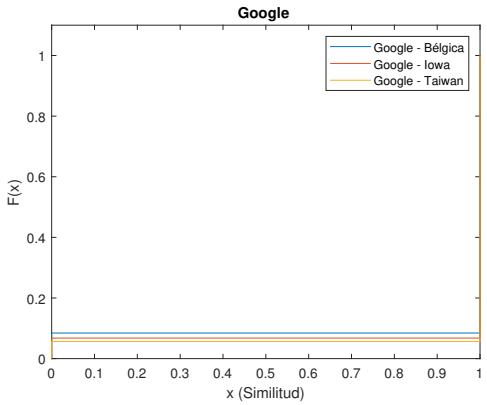
(a) Probabilidad del Cloud por inter-CSP con la métrica 1



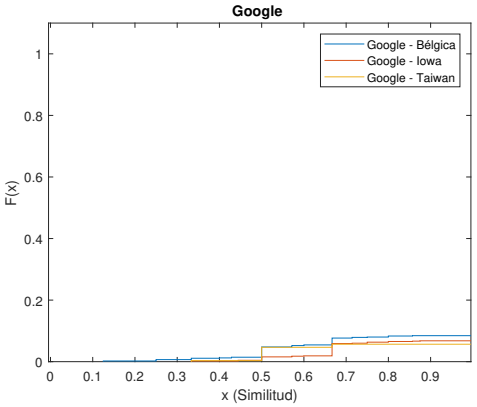
(b) Probabilidad del Cloud por inter-CSP con la métrica 2

Figura C.3: Probabilidad del Cloud por inter-CSP con respecto a la similitud de las rutas

CSP origen en la nube

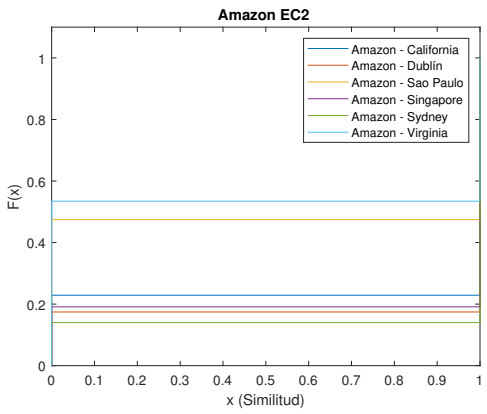


(a) Probabilidad del Cloud por CSP origen de Google con la métrica 1

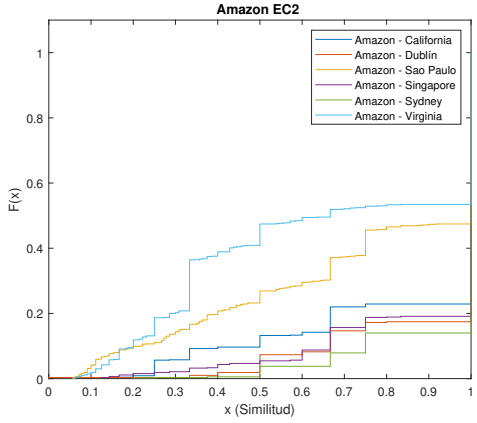


(b) Probabilidad del Cloud por CSP origen de Google con la métrica 2

Figura C.4: Probabilidad del Cloud por CSP origen de Google con respecto a la similitud de las rutas

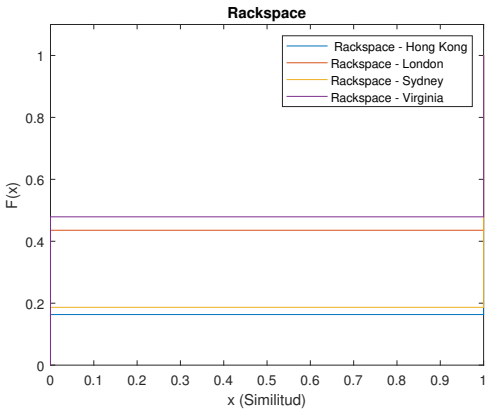


(a) Probabilidad del Cloud por CSP origen de Amazon con la métrica 1

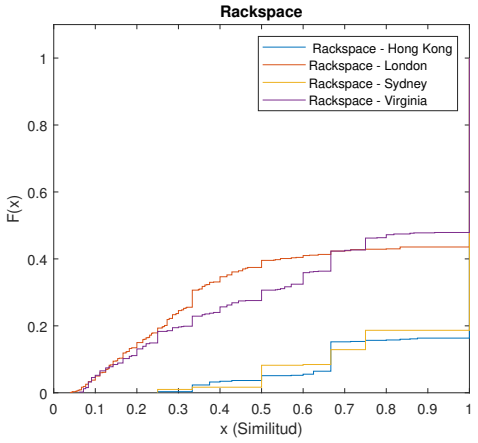


(b) Probabilidad del Cloud por CSP origen de Amazon con la métrica 2

Figura C.5: Probabilidad del Cloud por CSP origen de Amazon con respecto a la similitud de las rutas



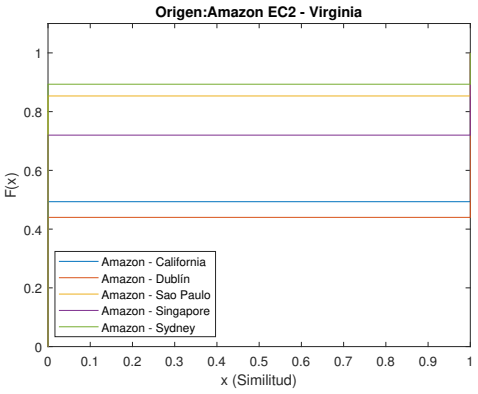
(a) Probabilidad del Cloud por CSP origen de Rackspace con la métrica 1



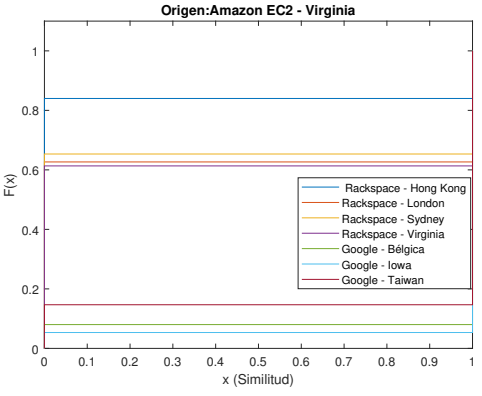
(b) Probabilidad del Cloud por CSP origen de Rackspace con la métrica 2

Figura C.6: Probabilidad del Cloud por CSP origen de Rackspace con respecto a la similitud de las rutas

Centros de datos origen en la nube

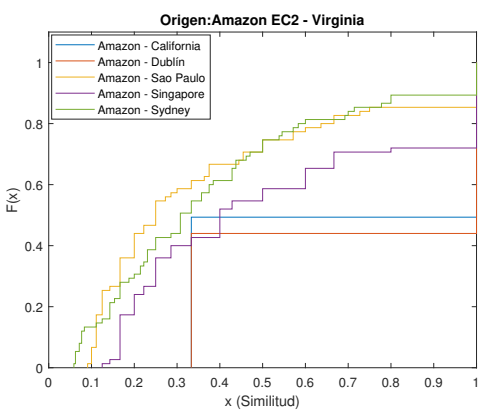


(a) Probabilidad del Cloud por DC origen contra destinos Amazon

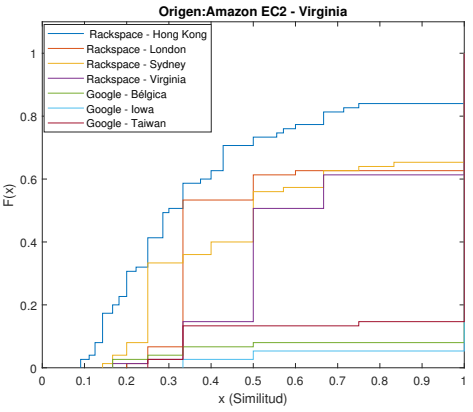


(b) Probabilidad del Cloud por DC origen contra destinos Rackspace y Google

Figura C.7: Probabilidad del Cloud por DC origen Virginia con respecto a la similitud de las rutas para cada destino con la métrica 1

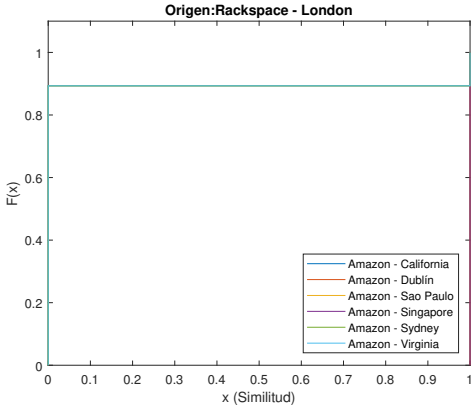


(a) Probabilidad del Cloud por DC origen contra destinos Amazon

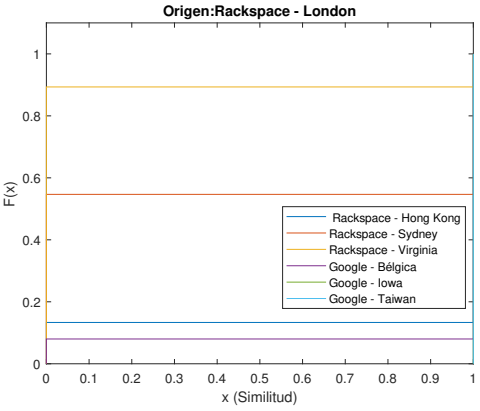


(b) Probabilidad del Cloud por DC origen contra destinos Rackspace y Google

Figura C.8: Probabilidad del Cloud por DC origen Virginia con respecto a la similitud de las rutas para cada destino con la métrica 2

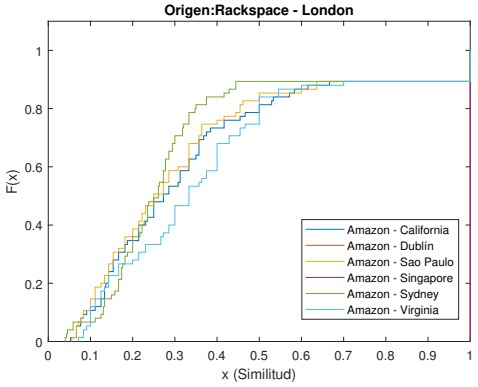


(a) Probabilidad del Cloud por DC origen contra destinos Amazon

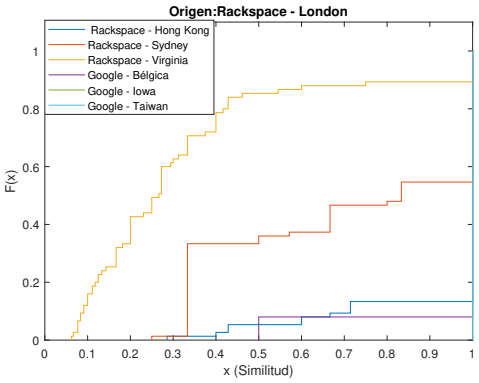


(b) Probabilidad del Cloud por DC origen contra destinos Rackspace y Google

Figura C.9: Probabilidad del Cloud por DC origen London con respecto a la similitud de las rutas para cada destino con la métrica 1

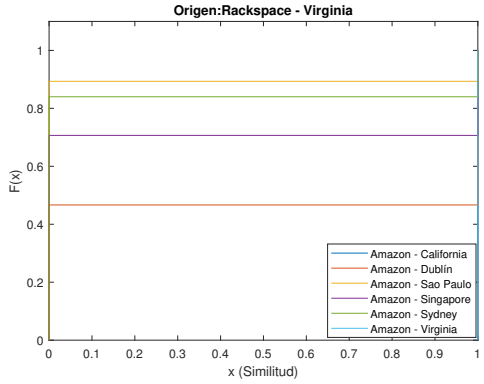


(a) Probabilidad del Cloud por DC origen contra destinos Amazon

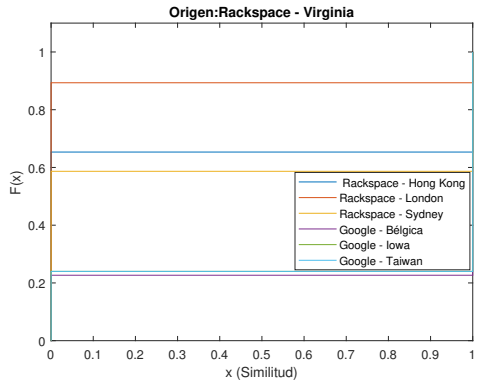


(b) Probabilidad del Cloud por DC origen contra destinos Rackspace y Google

Figura C.10: Probabilidad del Cloud por DC origen London con respecto a la similitud de las rutas para cada destino con la métrica 2

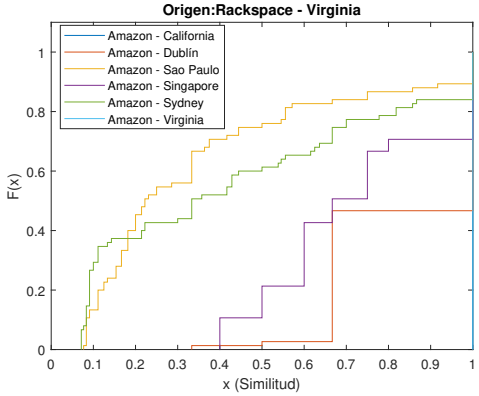


(a) Probabilidad del Cloud por DC origen contra destinos Amazon

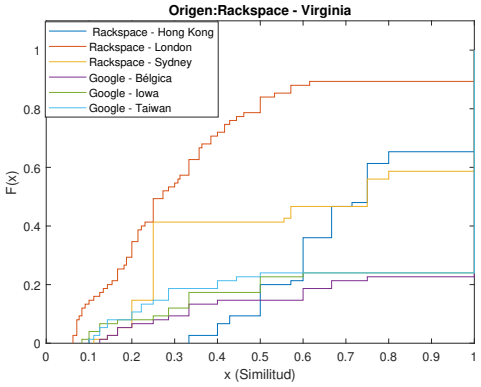


(b) Probabilidad del Cloud por DC origen contra destinos Rackspace y Google

Figura C.11: Probabilidad del Cloud por DC origen Virginia con respecto a la similitud de las rutas para cada destino con la métrica 1



(a) Probabilidad del Cloud por DC origen contra destinos Amazon



(b) Probabilidad del Cloud por DC origen contra destinos Rackspace y Google

Figura C.12: Probabilidad del Cloud por DC origen Virginia con respecto a la similitud de las rutas para cada destino con la métrica 2